DOCUMENT RESUME

ED 154 773 IR 005 648

AUTHOR Emery, James C.; And Others
TITLE Simulation and Gaming Project for Inter-Institutional Computer Networking. Volume I.
INSTITUTION Interuniversity Communications Council (EDUCOM), Princeton, N. J.
SPONS AGENCY National Science Foundation, Washington, D.C.
PUB DATE Jul 76
GRANT DRC75-03634
NOTE 189p.

EDRS PRICE MF-$0.83 HC-$10.03 Plus Postage.
DESCRIPTORS *Computer Oriented Programs; Educational Research; *Futures (of Society); Game Theory; *Higher Education; *Information Networks; Interinstitutional Cooperation; Models; *National Programs; *Simulation

ABSTRACT

The Simulation and Gaming Project for Inter-Institutional Computer Networking is a joint effort on the part of EDUCOM and 18 participating institutions to investigate the role that computing networks might play in higher education and research. Central to the project is the development of a computer simulation model of a possible national network, composed of the participating institutions, in which services can be exchanged through a market medium. This is a report on the results of the first year of the three year study. Included in this phase were the development of representational concepts, the design and implementation of the basic simulation model, the collection of data from the participating institutions, and the conduct of some preliminary experiments using the model. Later emphasis will be on using the model for a comprehensive investigation into the organizational implications of a network, the conditions necessary for a successful network, and the likely problem areas that must be monitored. (Author)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

ED154773

EDUCOM
IR005648

Report to the

NATIONAL SCIENCE FOUNDATION

On Year I of the Simulation and Gaming Project for Inter-Institutional Computer Networking

Volume I

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Donna L. Davis

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND USERS OF THE ERIC SYSTEM."

July 1976

Report to the

NATIONAL SCIENCE FOUNDATION

Simulation and Gaming Project for Inter-Institutional Computer Networking

Year 1

Grant Number DCR75-03634

Principal Investigator: James C. Emery, President, EDUCOM

Project Manager: Ronald Segal

Project Consultant: Norman R. Nielsen, Manager, Information Systems Group, Stanford Research Institute

Cooperating Investigators:

Robert Ashenhurst, University of Chicago
Sanford V. Berg, University of Florida
Donald L. Kreider, Dartmouth College
James R. Miller, Stanford University
K. Roger Moore, Texas Tech University
Joe B. Wyatt, Harvard University

Project Staff:

Steven Bensinger, Programmer, EDUCOM
Deborah Brown, Secretary/Administrative Assistant, EDUCOM
Paul Heller, Consultant-Data Collection and Benchmark Tests, EDUCOM
Beverly O'Neal, Systems Analyst, EDUCOM
Joseph Puglisi, Programmer, EDUCOM
Norman H. White, Consultant-Systems Design, New York University

James C. Emery

President, EDUCOM

EDUCOM
Interuniversity Communications Council, Inc.
Princeton, New Jersey 08540
(609) 921-7575

July, 1976

TABLE OF CONTENTS

		Page
ABSTRACT		v
I.	Executive Summary	
	A. Introduction	1
	B. Development of the Network Simulation and Gaming Model	2
	C. Characteristics of the Simulation Model	3
	D. Project Management	4
II.	Introduction	
	A. Background	7
	B. Objectives of the Project	11
	C. Project Phases	13
	D. Organization and Conduct of the Project	14
	E. Organization of the Report	17
III.	Simulation Model	
	A. Design Philosophy	23
	B. Model Overview	24
	C. Model Design	26
	D. Implementation Conventions and Procedures	29
	E. Simulation Run Requirements	38
	F. Model Documentation	40
	G. Validation	43
IV.	Results of Background Studies	
	A. Purpose and Approach	47
	B. Network Organization and Administration	48
	C. Representational Concepts	50
	D. Computer System Performance Modeling	61
	E. Site Representations	71
	F. Supply Determination and Estimation	74
	G. Demand Estimation	76
	H. Market	78
	I. Financial Considerations	78
	J. Communications	83
V.	Data Collection and Analysis	
	A. Overview	87
	B. Questionnaire #1	88
	C. Questionnaire #2	91
	D. Benchmarks	97
	E. Observations on Data	103

		Page
VI.	Phase I Experiments	
	A. Overview and Purpose	107
	B. Areas of Experimental Interest	108
	C. Conduct of Experiments	108
	D. Experimental Runs	109
	E. Summary of Findings	125
VII.	Summary of Project Status	
	A. Simulation Model	127
	B. Site Data	128
	C. Status and Implementation of Background Studies	128
	D. Phase I Simulation Experiments	132
	E. Perspective Relative to Work in Phases II and III	133
VIII.	References and Publications	135
APPENDICES		
I.	Model Overview and Flows	
II.	Model Policies and Representations	

(Volume II)

III.	Model User's Guide
IV.	Model Reference Guide
V.	Modifications Guide
VI.	Questionnaire #1
VII.	Questionnaire #2
VIII.	Benchmark Test Descriptions

(Volume III)

IX.	Listings

ABSTRACT

The Simulation and Gaming Project for Inter-Institutional Computer Networking is a joint effort on the part of EDUCOM and eighteen participating institutions to investigate the role that computing networks might play in higher education and research. Central to the project is the development of a computer simulation model of a possible national network, composed of the participating institutions, in which services can be exchanged through a market medium.

This is a report on the results of the first year of the three year study. Included in this phase were the development of representational concepts, the design and implementation of the basic simulation model, the collection of data from the participating institutions, and the conduct of some preliminary experiments using the model.

Later emphasis will be on using the model for a comprehensive investigation into the organizational implications of a network, the conditions necessary for a successful network, and the likely problem areas that must be monitored.

I. EXECUTIVE SUMMARY

A. Introduction

Among the most difficult questions confronting decision makers at colleges, universities and research institutions are those concerning the best manner of satisfying the expanding and increasingly varied demands for computing services. The vast increase in computer use now imposes serious financial burdens on many educational institutions, necessitating that they seek more efficient and effective ways of providing needed capabilities.

One of the most promising means of accomplishing this goal is that of sharing computer resources through a national computer network. Sharing is not just a matter of economy, but it can open up new possibilities to the educational and research community. It has the potential to offer to all institutions throughout the country the best computing resources available -- a variety and quality of resources which not even the largest single institution could hope to provide on its own.

There is very little doubt that such sharing will take place, since it already exists in at least a relatively primitive form. However, the extent and form of the sharing will depend on many complex organizational, economic, and behavioral issues. Even if a network only focuses on the needs of higher education, a large number of alternative arrangements are possible. For example, the degree of centralization of the network, pricing schemes, and budgeting procedures at each university all introduce possible variations. An institution's policies with respect to outside purchases, and its willingness to assume the risk (and gain the benefits) of developing the resources required to become a network supplier, will also affect the network behavior.

Network arrangements and institutional policies interact in complex ways. It is not possible, therefore, to make reliable a priori predictions about the consequences of the many possible combinations of alternative decisions. This is true of macro decisions that affect the entire network, as well as at the micro level of an individual institution or an individual user.

It is extremely difficult to experiment with a functioning network, except in only minor incremental ways. Analytical models, while useful in providing insights into network phenomena, cannot begin to capture the full richness of possible behavior that would interest network designers and institutional policy makers. Of the tools available, then, only simulation techniques permit investigation of the full range of alternatives that must be considered.

B. Development of the Network Simulation and Gaming Model

The current simulation project grew out of a six-month[1] planning study funded by the National Science Foundation (NSF). A group of eight "cooperating investigators," representing a wide variety of academic backgrounds, planned and recommended support of a major effort to investigate the role that computing networks might play in higher education. The group felt that a simulation model would be a necessary tool of such an investigation. EDUCOM submitted a proposal to perform the study, and in February, 1975 it was approved by NSF.

The project extends over a three-year period. This first phase encompassed many areas including the development of representational concepts, the design and implementation of the basic simulation model, and the conduct of initial experiments. It also included the collection of data from 16 educational and research institutions that are participating in the network experiments. The collected data describe -- albeit, in relatively preliminary form -- each institution's present demand for, and supply of, computing services. Collectively the 16 institutions constitute

a possible network in which services can be exchanged through a market mechanism.

The second phase of the project, which has already begun, will concentrate on obtaining a deeper understanding of each participating institution's facilities, computing demands, and policies affecting its network activities. The results of these studies will be reflected in refinements to the model so that it can more faithfully represent the specific behavior of each site on the network. It will then be used to examine the effect of a variety of behavioral, decision, and policy patterns on networks and institutional members.

The final phase calls for a modification of the model to permit human decision makers to input decisions interactively during a simulation run. In the current Phase I version of the model, decisions are made automatically by the model based on built-in policy rules. In the "gaming" version, administrators at the participating institutions will be able to modify decisions at any point in simulated time. In this way an administrator can make ad hoc decisions in any arbitrary way, rather than being forced to define a rule that remains in effect throughout the simulated run. Thus the gaming phase will introduce a reality to the model that could not be achieved solely with pre-defined rules.

C. Characteristics of the Simulation Model

Although construction of the simulation model is only one part of the overall project, it is a vital part. The model provides an essential tool for the studies that constitute the real justification for the project.

The model was constructed in a modular, top-down fashion. This has permitted testing of its higher-level components as development work proceeded. This approach also offers the tremendous advantage of flexibility: lower-level modules can be added or modified without affecting other modules or the overall structure of the model. This

is an absolutely necessary requirement for a model that will undergo continual evolutionary development over the life of the project.

When the model is run, time moves forward in fixed weekly increments. During each weekly calculation, demand for computing services is generated according to the built-in rules (or, during the gaming phase, to any ad hoc changes made by a human decision maker). The rules take into account policy decisions and the status of the network during the previous weekly time interval (e.g., response time at each computer center, cost of each service, etc.). The aggregate demand placed on a given site depends on the demand throughout the network, policy constraints on network purchases, physical constraints on communications capacity, and the attractiveness of the site relative to other sources of supply (including, of course, a user's own local computer center).

Time moves forward week-by-week in this fashion. Demand and supply at each institution and the flow of services among centers shift dynamically as simulated events unfold. Periodic reports and final summary reports are produced to describe shifting supply and demand levels, the flow of work, and selected financial variables at each institution.

The model is written in FORTRAN to make it as transportable as possible. Special care has also been given to careful documentation to increase the model's transportability. Participating institutions, as well as the academic and research community at large, will be encouraged to use the model in exploring alternative network decisions or organizational arrangements.

D. Project Management

The model is large enough, and involves a large enough group of participants in its development, that careful attention had to be given to management of the project. From its inception, the project called for the coordination of researchers drawn from a number of different institutions. This required close attention to

documentation and dissemination of the current studies of the project.

The cooperating investigators have continued with the project throughout its life. Periodic review meetings have been held to critique the work done by the EDUCOM staff and to obtain further advice for future development work.

Representatives from all of the participating institutions have attended one of three regional meetings that were presented. The purpose of the meetings was to establish personal contact with each institution, to explain the model, to describe data collection requirements, and to obtain feedback comments from participants. .

A number of special studies have been commissioned as an integral part of the project. One of these focused on developing a perception of computing services that flow across the network. Another developed a model for predicting the impact on a computer center of shifts in demand for its services. Similar studies focusing on program benchmarking, network marketing and its effect on demand, and pricing and budgeting strategies have been commissioned and will play an important part in enhancing the value of the research. The intent of this approach is to draw upon the best resources available to help in developing the model, while still retaining overall project coordination in the hands of the EDUCOM staff.

The emphasis in Phase I was on the research and background work necessary to design and implement the basic model, and on the use of this model to study factors influencing network behavior. As a result of the successful completion of this work in Phase I, efforts can proceed with the application of the model to the more interesting behavioral, organizational and impact issues which will be examined in Phases II and III.

The project is basically adhering to the schedule and budget established in the original proposal. Some parts of the project are proceeding somewhat faster than expected, while other parts are proving more difficult and time consuming than planned. For example, the modular construction of the model will reduce the expected effort required in Phases II and III to adapt it to individual institutions and to incorporate interactive modification of decision rules. Data collection, on the other hand, is somewhat behind schedule. The expectation is, however, that the overall project will be completed within the proposed time and budget.

II. INTRODUCTION

A. Background

Decision makers at educational and research institutions are grappling with difficult questions about how to best satisfy the expanding and increasingly varied demands for computing services. It is clear that, although user demands will continue to grow, the cost spiral must be controlled. Technological advances in micro- and mini-computers, and in large scale hardware facilities, will not, in themselves, be sufficient to economically satisfy the requirements for accessibility to a greater variety and sophistication of service offerings.

Networks offer one of the most attractive possibilities for improving the efficiency and effectiveness of computing[2]. The report of the deliberations at a NSF-sponsored series of General Working Seminars on computer networking[3] held in 1972 and 1973 indicated that it is now technically feasible to create a network linking research and educational computers at colleges, universities, and research institutes. Although some technological problems remain, the difficulties of linking these institutions are primarily nontechnical in nature. In particular, major economic, political, and organizational considerations are likely to pace the development of a successful network. These issues can successfully be dealt with only on the basis of a clear understanding of the potentials, limitations, and implications of networking.

Existing networks provide a good indication of the costs and technical characteristics of a national network. It is still not clear, however, how a dynamic national networking environment would impact a given institution in terms of economic, organizational, political, and intellectual effects. Nor is it known how these effects would vary with the particular computing philosophy, policies, and practices in effect at an institution. For example, what changes would a network bring in instruction, research, science

information, and administrative computing? What would be the impact of "balance of payments" problems? How would users be affected by the availability of multiple types of resources at a variety of prices? What changes would take place in the types of computing resources developed or maintained by an institution? What impact would a network have on established institutional policies?

The characteristics of a network and the ways it might evolve in the face of collective institutional choices also need to be examined. What types of decisions or policies relative to the network must be established by university resource managers, users, and top-level decision makers? To what extent and in what areas is centralized management of the network required? What contractual arrangements should be made among buyers, suppliers, and network administrators? How should prices be set? How should invoices for services rendered be handled and accounts paid?

These questions can be examined only in an environment in which the implications of such policy questions can be observed in concrete terms. Moreover, that environment must not be constrained by a priori solutions to these issues. In particular, questions of network management and control must be left open.

No existing network has the necessary characteristics to examine all of the important issues. The ARPA Network, for example, is devoted to Department of Defense activities. Thus it is not suitable as a basis for studying the implications of an economically viable general network linking diverse educational institutions and a wide spectrum of users.

The use of a real network to examine these issues, while advantageous in some respects, poses a number of almost insuperable problems. It would be extremely costly, take several years to implement, severely restrict the number and scope of approaches and alternatives that could be investigated, disrupt the normal operation of the network, and require significant commitments of

intellectual and other resources. Consequently, such an approach would not be considered without an overwhelming demonstration that the likely benefits would justify the costs and risks of experimenting with a real network. With present knowledge, no such demonstration is possible.

It was in recognition of these difficulties that a simulation approach was taken to the study of a national network. The objective was to develop a model of a computer network to represent the conditions that might prevail in a real networking environment. The simulation approach permits an effective exploration of the potential impact upon an institution of participating in a network, as well as an examination of the ways in which that environment is affected by the decisions and policies of participating institutions. It provides flexibility in the types of networking situations that can be considered and allows for the testing of a variety of alternatives at a relatively low cost.

The development of a large simulation model is a complex and difficult undertaking and must be carefully planned if it is to be effective. In order to do such planning EDUCOM* brought together a group of eight individuals knowledgeable in the areas of model building, gaming, economics, resource administration, and educational computing. After an initial feasibility study, NSF support was secured for an intensive planning study (NSF grant GJ-41429). Over a six-month period, assisted by the EDUCOM staff, these individuals established the specific goals and objectives for such a simulation and gaming project, outlined the data to be collected, selected the initial 16 participating institutions, determined the level of detail and framework of the basic network model, and prepared a detailed plan[1] for the actual conduct of the project.

*EDUCOM is a consortium of more than 200 colleges, universities, and non-profit organizations that serve higher education. It was founded to help its members make the most effective use of computer and communications technology.

A proposal[4] to implement that research plan over a three year period was submitted to the NSF in September, 1974. A grant (DCR75-03634) to support the first year's efforts on this project was awarded by NSF in February 1975. This report documents the results of that work.

In order to provide a basis upon which the details of the simulated network could be constructed, a group of 16 institutions -- being augmented to 18 -- has been cooperating in the project. The participating institutions are as follows:

Bryn Mawr College	National Bureau of Economic Research
Carnegie-Mellon University	Ohio State University
University of Chicago	University of Pennsylvania
Dartmouth College	Saint Olaf College*
University of Georgia	Stanford Research Institute
Harvard University	Stanford University
University of Iowa	Texas Tech University
Lehigh University	University of Texas
Massachusetts Institute of Technology	Vassar College*

*These two institutions expect to join the project by September 1976.

The institutions were selected on the basis of willingness to participate and their ability to make a positive contribution. Collectively they provide a wide range of institutional sizes, missions, sources of funding, unique computer facilities and services, and networking experience.

The participating institutions are providing the basic data needed to exercise the network model. The model, in turn, is being used as a research tool to further the knowledge of the networking process. It will be used in a gaming environment during the last phase of the project to help decision makers at participating institutions explore the potentials of networking for their institutions, and the implications that their own attitudes, computing policies, and local policies might have on the network.

B. Objectives of the Project

The project provides a simulated environment in which two principal objectives can be pursued. The first objective is to explore the parameters that govern network behavior and to isolate and examine those elements critical to the success or failure of a network. The second objective is to help institutional decision makers develop an understanding of the impact of a national network on their internal resource allocation process.

The first objective requires investigation of a number of issues critical to the behavior of a network. These can be classified into policy issues and structural issues. Policy issues include pricing, funds flow, network standards, service guarantees, user support, marketing, and capacity adjustment. These issues are all closely tied to the various alternative structures for network management. Accordingly, several structures have been hypothesized, spanning the spectrum from a loose set of independently initiated bilateral agreements between individual institutions, to a highly structured and centrally managed network. Each of the policy alternatives will be examined in the light of the various suggested network structures.

The second objective of the study is to improve each institution's ability to make decisions and establish policies pertaining to networks. Decision makers need clear insights about the implications of their policies, as well as the implications of network actions on their own institution. Gaining such insights in the complex real world can be very difficult and expensive.

Some of the data already collected indicate that institution administrators often do not have sufficient contact with networks to have established clear policies governing network situations. As a consequence, policies, where they exist, are often incomplete and inconsistent and may contain unanticipated implications for the institution or for the network. Feedback to administrators

from the model-based investigations should clarify the present intuition-based conceptions about the impact of their policies. The model will also be employed to study institutional policy positions and to assess the likely advantages and disadvantages of various modes of network participation. Working with the project is thus aiding administrators in obtaining a much clearer understanding of appropriate network policies for their institutions.

Several participants in the network simulation project have also expressed interest in gaining information about the possible markets for, and the potential external support requirements of, specialized resources that they might offer on a network. The derivation of this type of information from the study should assist these institutions in planning their computing activities. Other institutions are presently engaged in bilateral resource exchange agreements and need more information about the likely implications of wider resource sharing. A third group wants to explore the possible implications of reducing or eliminating selective in-house services in favor of outside suppliers. Finally, and perhaps most numerous, are the institutions that have a strong need for limited access to sophisticated facilities and services whose supply cannot be justified internally. Many small colleges (and even some very large universities) throughout the country would fall into this last category.

An important by-product of the study will be the derivative impact upon the institutions that participate in the project. Administrators, users, and computing center directors are collecting data about their computing activities and examining them in detail in cooperation with the project staff. This experience appears likely to influence the attitudes of decision makers about the ways in which networking can be applied effectively in support of research and education.

A clearer understanding of network behavior will be useful

not only to policy makers at each institution, it will also be valuable to a wider group interested in networks. For example, Federal policy makers who are concerned with the effective utilization of the nation's computing resources are likely to benefit from a close scrutiny of network behavior. Computer scientists and other researchers having an intellectual interest in networks can use the model to explore various hypotheses about networks.

The programs for the simulation model and gaming study will be made publicly available. This will allow institutions and researchers outside of the project to conduct their own studies and use the model for improving their decision making. Considerable effort is being made to make the computer programs as modular as possible in order to foster such use.

C. Project Phases

The overall project is broken down into three overlapping phases. The first phase -- the subject of this report -- resulted in development and use of a simulation model. The second phase will focus on decision making at the individual participating institutions. It calls for tailoring the model to each institution through refinement of the data that describe capacities, demands for computing, management policies with respect to computing, and the like. The final gaming phase calls for administrators at each participating institution to make decisions in a simulated network -- i.e., to set policies as events unfold to them in simulated time.

Phase I efforts have been primarily concentrated on initial data collection and the design and programming of a computer simulation model. Design of the model adheres closely to a top-down structure. Thus, the model consists of a hierarchy of modules, with each module relatively independent of the others. A module is kept fairly small and is confined to a limited and well-defined task. This approach permits modification and evolution of the model, since it allows extensive modification or replacement of a module without affecting other modules and without changing the overall structure

of the model.

The model has been used during the last part of Phase I for exploratory simulations to investigate a variety of possible site and network practices and situations. The objective has been to isolate and examine those parameters that are critical to network development and success.

Phase II began before completion and documentation of the Phase I simulation analyses[5]. Focus during the second phase will be on the determination of the actual policies and practices of the participating institutions and on the insertion of these results into the model. The implications of each institution's actual policies and practices, on both the institution and the network, will be reviewed with the participants to determine the reasonableness of the decision rules used in the model and plausibility of the simulated results. The decision rules will be refined as appropriate through an iterative process.

Phase III will be initiated in the second project year (in parallel with Phase II) and will continue until the end of the third year. The major orientation of Phase III will be toward a network simulation analysis made in a group gaming environment. Participants will dynamically make decisions and alter policies based on the reflected implications of earlier decisions. Thus, they will have to "live" with the consequences of their decisions.

D. Organization and Conduct of the Project

The simulation and gaming project has been conducted as a cooperative effort from its very inception. The original planning study grew out of a number of early discussions organized by Henry Chauncey, then President of EDUCOM, and several networking authorities. In order to formalize widespread participation, a group of eight "cooperating investigators" were drawn from various educational institutions to serve in a continuing advisory capacity throughout the life of the project. The cooperating investigators have proven

to be extremely helpful in generating creative ideas, cautioning against possible pitfalls, and providing feedback response to work done by the research staff. Membership in the cooperating investigators panel is as follows:

Dr. Robert L. Ashenhurst
Director, Institute for Computer Research
University of Chicago

Dr. Sanford V. Berg
Assistant Professor
Department of Economics
University of Florida

Dr. Donald Kreider
Professor
Department of Mathematics
Dartmouth College

Dr. James R. Miller
Associate Dean
Graduate School of Business
Stanford University

Mr. K. Roger Moore
Texas Tech University

Dr. Charles H. Warlick
Director, Computation Center
University of Texas at Austin

Mr. Joe B. Wyatt
Vice President for Administration
Harvard University

Much of the detailed work during the earlier planning studies was conducted by Dr. Norman Nielsen of the Stanford Research Institute. Dr. Nielsen's experience as a computer center manager and his research on networks and resource allocation uniquely equipped him to provide technical leadership for that project. Following the completion of the planning study, he has continued as project consultant and provides frequent advice and assistance to the EDUCOM project staff.

Dr. James C. Emery, President of EDUCOM, serves as Principal Investigator of the project and is responsible for its overall direction. Mr. Ronald Segal, of the Graduate School of Business at New York University, contributes much of the technical leadership, as well as day-to-day administration of the project. Ms. Beverly O'Neal serves the project on a full-time basis as systems analyst. During Phase I, Dr. Norman White of NYU and two outstanding NYU students, Mr. Steven Bensinger and Mr. Joseph Puglisi, have provided analysis and programming assistance. A full-time secretary/administrative assistant, Ms. Deborah Brown, has provided valuable support. The staff of another EDUCOM activity, the Planning

Council[6] on Computing in Education and Research, works closely with the Simulation and Gaming Project. (The Planning Council is a joint activity of twenty-two schools with a mission to explore and implement the use of networks to share computing resources. Not all Planning Council institutions also participate in the Simulation and Gaming Project).

Much of the detailed data collection for the project is provided by personnel from the participating institutions. Each institution contributes the time of a senior administrator, the director of computing activities, and a "liaison coordinator." A research assistant is supported by project funds to assist the liaison coordinator in collecting data for input into the model. The individuals that served in these functions during Phase I are shown in Figure II-1.

Three regional meetings were held with institutional representatives during the fall of 1975. The first was held in Palo Alto, the next in Cambridge, and the final one in Chicago. The purpose was to meet the institutional representatives, to discuss the current state of the model, and to outline the role that the institutions were expected to play. The meetings proved to be invaluable in removing some of the ambiguities in the model design and data collection questionnaires. They also provided a very useful forum for general discussion and gaining the benefit of the vast and varied experience of the participants.

To the extent possible, the project has utilized existing resources available from the academic and computer science community, rather than attempting to build its own staff to duplicate existing capabilities. Consistent with this philosophy, several well-defined tasks have been contracted for with academic institutions and other technical organizations. Dr. Norman Nielsen and some of his colleagues at Stanford Research Institute (in particular, Dr. Clifford A. Isberg and Dr. A. Robert Tobey) have worked on several of the more challenging technical problems -- principally the definition

of computing service types and their translation into computing resource requirements. Dr. Jeffrey P. Buzen of BGS Systems Inc. (who is also affiliated with Harvard University) has applied his work in network queueing models to the modeling and prediction of performance on a computer system under varying workloads. Both of the tasks are critical to the overall project, and both organizations bring to bear on their assigned problems experience and skills that would be very difficult (if not impossible) to duplicate.

Given the decentralized nature of the project, it has been essential to provide central coordination and guidance[7]. This has been achieved through a meticulous specification of systems design and programming standards, careful documentation combined with dissemination of information to participating groups, and frequent technical reviews of the project with EDUCOM staff members and principal consultants.

E. Organization of the Report

This report is designed to be a self-contained document that provides a comprehensive and detailed description of the current status of the Simulation and Gaming Project. It is organized hierarchically, getting into increasing detail as the discussion proceeds.

Following the overall summary contained in this introductory section, further elaboration is given to the model in Section III. Special emphasis is placed on the design philosophy and the motivations for taking the approaches that were followed.

Much of the detailed investigation connected with the project was broken down into a series of "background studies" that could be done in-house or assigned to decentralized groups. These studies are summarized in Section IV.

A major activity of the project was the collection of data about computing activities in each of the participating institutions.

These data have been collected primarily through the means of detailed questionnaires and benchmark programs. Section V discusses the issues involved in data collection, describes the instruments used, and points out some of the difficulties encountered.

The model is currently being used to conduct a series of experiments to gain some preliminary insights into network behavior and to identify the more critical components of the model. These experiments are discussed in Section VI.

Section VII summarizes the current status of the model and other aspects of the project. In addition to this, it outlines the continued work during Phases II and III.

The appendices give much more detailed information for those who want to gain a deeper understanding of the model or plan to use or modify the model. Appendices I and II contain material of fairly general interest and are included in Volume I, along with the more general material contained in the main text (Sections I through VIII).

Volume II contains the appendices of interest primarily to those who intend to use the model, or who are interested in reviewing the data collection or benchmarking procedures. It defines run concepts and outputs, and describes the actual operation of the model. Appendices III, IV and V represent the model user's guide, model reference guide, and detailed instructions for making programming modifications. The remaining appendices contain reproductions of the two data questionnaires, as well as the instructions and listings of the benchmark programs used.

Volume III is available upon request and gives still more detailed information for those needing the actual program listings of the entire system in order to modify or extend the model.

Figure II-1

Network Simulation and Gaming Project

Institutional Participants

Institution	Administrator	Computer Ctr. Director	Liaison Coordinator	Research Assistant
Bryn Mawr College	Mr. Robb Russell (Acting)	Mr. Robb Russell Director of Computing Services	Mr. Robb Russell	Mr. Robb Russell
Carnegie-Mellon University	Mr. Richard Van Horn Vice President for Business Affairs	Dr. John W. McCredie Vice Provost for Information Services	Mr. Peter Wolk Assistant Director for Special Projects	Mr. Peter Wolk
The University of Chicago	Dr. D. Gale Johnson Provost	Mr. Fred H. Harris Director, Computation Center	Mr. George R. Bateman Senior Staff Analyst, Computation Center	Mr. George R. Bateman
Dartmouth College	Dr. Thomas E. Kurtz Director, Office of Academic Computing	Mr. John S. McGeachie Director, Computing Services	Mr. Eugene A. Fucci Assistant Director - Special Projects, Kiewit Computation Center	Mr. Eugene A. Fucci
The University of Georgia	Dr. James B. Kenney Associate to the Provost	Dr. James L. Carmon Director, Office of Computing Activities	Miss Margaret K. Park Director of Information Services, Office of Computing Activities	Ms. Carolyn Guard
Harvard University	Mr. Robert H. Scott Director, Office of Information Technology	Mr. Guy Ciannavei Harvard University Computing Center	Mr. Eric Lentz	Mr. Eric Lentz
The University of Iowa	Dr. D. C. Spriestersbach Vice President and Dean of Graduate College	Dr. Howard Dockery Director of Computer Center	Mr. Chuck Shomper Director of Regional Computer Center Dr. Gary Wicklund Professor of Business	Mr. Harland Garvin Manager of D.P. for Networks
Lehigh University	Dr. Joseph F. Libsch Vice President for Research	Dr. John E. Walker Director of Computing Center	Dr. John E. Walker	Ms. Judith A. Swartley
Massachusetts Institute of Technology	Mr. Weston J. Burner Director of Information Processing Services	Mr. Joseph R. Steinberg Associate Director of Information Processing Services	Ms. Brenda L. Ferriero Computer Services Coordinator	Ms. Brenda L. Ferriero

Figure II-1 (cont.)

Institution	Administrator	Computer Ctr. Director	Liaison Coordinator	Research Assistant
National Bureau of Economic Research	Warren C. Lacksrom Assistant Vice President, Finance and Administration	Gerald Ruderman Computer Operations Director	Ms. Helen B. Munzer Coordinator, Computer Operations	Mr. Walt Maling
The Ohio State University	Albert J. Kuhn, Provost Office of Academic Affairs	Dr. Roy F. Reeves Director, I & R Computer Center	Mr. Marion L. Tripp Assistant to the Director I & R Computer Center	Badie Farah Graduate Research Associate
University of Pennsylvania	Paul O. Gaddis Vice President for Management	Dr. James Niederer	Dr. James Niederer Associate Director, Computing Activities	Dr. Roger Warburton Research Associate in Physics
Stanford Research Institute	Mr. Harvey L. Dixon Vice President, Finance and Administration	Mr. Vincent T. Lauricella Manager, B6700 Computer Center	Dr. Norman R. Nielsen Manager, Information Systems Group	Ms. Pam Antiscevich Research Assistant, Computer Center
Stanford University	Dr. William F. Massy Vice Provost for Research Dr. Gene F. Franklin Associate Provost for Computing and Professor of Electrical Engineering	Charles R. Dickens Director of the Stanford Center for Information Processing	Ms. Patricia L. Devaney Assistant to the Vice Provost for Research	Mr. Marshall Ball
Texas Tech University	Dr. Monty Davenport Senior Associate Vice President	Herman Phillips Manager, Information Processing Services	Dr. James B. Wilcox Associate Dean for Research	Mr. Ben Ayres Teaching Assistant
University of Texas	Dr. Eldon Sutton Vice President for Research	Dr. Charles H. Warlick Director, Computation Center	Dr. James C. Browne Computer Sciences Department	Dr. G. Scott Harris

III. SIMULATION MODEL

It should be made clear that this is not a project to build a simulation model; rather, it is a complex research activity seeking to answer some very critical questions relative to the future of national computer networking for research and educational institutions. The simulation model provides an essential tool to accomplish these objectives. By means of the model, it will be possible to examine the likely consequences of a wide range of alternative network policies. Although most of the effort during Phase I has been devoted to the design and implementation of this model, future efforts on the project will focus on using the model to accomplish the required objectives. This section describes the design and characteristics of the initial model.

A. Design Philosophy

The basic philosophy of the design is to provide a highly parameterized and flexible model. In general, it is "policy-driven" -- that is, its execution and the outputs it produces depend on the particular choice of policies used in a given run of the model. The objective is to permit the examination of a variety of institution and network policy rules in order to study the impact of various network configurations, management structures, usage modes, and growth patterns. Virtually every module starts with the input of appropriate policies, practices, and/or management decision. Technical relationships are then used only to the extent required to reflect adequately the implications of these decisions.

The model has been designed and implemented using a modified top-down, structured programming approach. The results of this effort tend to support the economies and efficiences in programming, as well as improvements in reliability, usually claimed by advocates of these techniques. Considering the size and complexity of the system, it was implemented in a comparatively short time with a small staff. Even though most of the programming was done by relatively inexperi-

enced students, there were no major debugging or validation problems. In addition, the modularization and clean definition of functions have permitted the segmentation of work so as to take full advantage of the variety of researchers and other personnel available to the project.

Perhaps the primary benefit from the top-down approach has been the ability to implement a useful working model quickly, even though some of the detailed modules remained in relatively crude form until the background studies and data collection were completed. Hence, early experience and insights were gained in such areas as work flow, output reporting, parameter tuning, and smoothing of time-varying estimates. This approach will be of continuing value as experience permits an increasing sophistication of representation in some of the more critical "lower level" modules.

Finally, mention should be made of the open-ended way in which representations of policies and practices are included. In general, each policy module calls subroutines representing the various required policies. A user of the model can therefore describe his site's practices and behavior by using any combination from a stored library of policies. In future project phases, users will be able to specify any desired ad hoc representation -- either by adding their own subroutines to the library or, in some cases, by entering decisions on-line. Considerable flexibility will thus be available in representing a given institution, and it will be relatively easy to modifiy or to expand the internal policy library on an on-going basis.

B. Model Overview

For purposes of model design, the "network" has been defined as having 20 initial sites. Eighteen of these are set aside for detailed representations of the 18 participating institutions on the project. (Sixteen were original participants and two have recently joined the project.) The actual number of sites used in any given run is an input variable, and most testing is being carried

out with smaller numbers of sites. One of the remaining two facilities has been designated as a "background" site that generates all work originating from outside the 18 members and receives all work specified for locations other than one of the member sites. This artificial site was introduced to represent the characteristics of a full mature network. The second extra site, referred to as a "network" site, permits the testing of network policy issues and special service categories. For example, requests to obtain a particular widespread service at "lowest cost" or "fastest turnaround" (i.e., without designation of a specific supplier) can be sent to this "site" for allocation to the appropriate supplier.

The description of each site in the network contains various policy formulations and decision rules. These deal with such matters as pricing, hardware changes, budget allocations, user support levels, and computer scheduling and priority setting. The model is designed in such a way that individual policy subroutines can be written and inserted to accommodate those sites that cannot be described by combinations of standard policies. At present, most of the site behavior and policy descriptions have been selected by the project staff. Although the choices were generally reasonable, the primary purpose for this phase was to exercise the model and to conduct some general sensitivity and trend experiments. During Phase II of the project, emphasis will be placed on formulating and specifying those options that actually describe the unique behavioral reactions, practices, and constraints of each site.

The design of the simulation model has three basic conceptual elements: supply offerings and capacity, user demand, and the balancing of desired demand with available supply ("market"). Each site on the network is viewed as a node having specified offerings and available capacity measured in terms of CPU speed, primary memory size, card reading and line printing potential, input/output channel capability, on-line ports, and communications bandwidth.

The demand for, and supply of, computing resources at each site is expressed in terms of categories of computing called "service

types." Each service type is presumed to include a reasonably homogeneous type of work. A very simple model might have only two, batch and interactive. The forty-eight present service types available should be sufficient to represent adequately the network resources. More could certainly be added, but memory requirements would grow proportionately and would rapidly exceed the space available.

The model operates with a basic time increment of one week. Although a week-long time increment precludes use of the model for investigating hour-by-hour variations in the processing loads at individual facilities, it does not imply that service characteristics having a time aspect of less than a week are ignored (e.g., shift or priority time differentials). It was concluded that time intervals of less than one week would be computationally prohibitive and that input data for smaller time increments would be unreliable. On the other hand, a weekly time increment is small enough to reflect overall network dynamics, and to be compatible with typical weekly, monthly, quarterly, and annual decision cycles.

The main time-varying information in the model is held in main storage in a three-dimensional matrix that contains the amount of each service type requested at each site on the network by every other site on the network. This matrix has a size of NSITES*NSITES* KTYPE -- where NSITES is the number of sites on the network (including the "background" and "network" sites) and KTYPES is the number of different (unique) services offered. The values of individual elements in the matrix are updated each simulation period (i.e., weekly).

C. Model Design

The model contents and operation will be illustrated by describing the program flow using a few of the top-level diagrams as an example. Implementation has been carried out by successively expanding and detailing the figures shown in this section, which are actual working diagrams. A more detailed analysis can be found

in Appendix II, which also includes a full set of flowcharts. This documentation can be used in hierarchical fashion, allowing the reader to penetrate into as much detail as desired along any path.

Figure III-1 shows the five major system modules. The flow of program control is from top to bottom, left to right. Thus, the entire system is executed by entering "SIMRUN" at the console. This executive procedure invokes module NETSIM, which sequentially calls the modules INPUT, ZSETUP, etc., as required. Looping is illustrated by the crosshatching in module 3.0. The symbol ∀ in that block should be read "for every," so that the module PROCES indicates a loop over all time periods (i.e., weekly).

Module 1.0, INPUT, begins with a console dialog to determine the basic conditions of the simulation run desired -- number and identification of sites, number of periods to be simulated, network structure, and the like. Additional comments to appear on the run output, such as data used or the purpose of the run, may also be included. Appropriate files are opened, and the basic data are read as required.

Those calculations that must be accomplished before the period-by-period looping are completed in ZSETUP. This includes determination of smoothing constants, conversion of raw input data into forms appropriate for later calculation, and output of a full set of test reports reflecting the system status at time zero (in the same form as the later test reports to be generated at weekly intervals).

Most of the actual processing takes place under control of the module PROCES. This includes all calculations necessary to represent the functioning of the network on a weekly basis and to provide the desired weekly reports.

The final two modules, COMPUT and GENREP, do the summary computations and reporting necessary to represent site and network behavior as a function of time. Included here are such areas as communi-

cations load, capacity growth, cash flows, and network utilization.

Module 3.0 in Figure III-1 is expanded in Figure III-2 to provide an overview of the weekly processing sequence. In each time period, the model sequentially handles all exogenous changes, supply determination, network demand estimation, and the balancing of supply against demand (market analysis). Period analyses are then performed and all required period reports are generated. Each of the indicated modules is entered sequentially as follows:

1. XOGEN - Exogenous changes are defined as any variable changes or policy descriptions that cannot be handled analytically in other modules. If such a change is to be made, it is entered here and put into effect directly. These changes override the variables and quantities that are otherwise developed analytically in later modules. Permissible changes include sites being added/dropped, major hardware changes at a site, revised policies or practices at a site, and changed network parameters. Eventually, in the gaming mode (Project Phase III), this module will be enhanced to permit on-line control over virtually every model parameter and policy.

2. SUPLY - Current computation capacity and offerings are determined in this module. These include (Figure III-3) supply policies and practices, budget and financial constraints, hardware and systems software, service offerings, prices, and level of support services.

3. DMAND - Demand estimates at each site are generated (Figure III-4) in a multi-step process. After the overall policies on demand are evaluated (3.31), estimates of demand for each user category at the site are determined (3.32). (User categories (Section IV.G.2) are defined to be relatively homogeneous aggregations of users at a site, such as students or funded researchers.) The "base" level of demand for a given user category is determined as a function of its most recent demand and site growth. This initial estimate is then modified by seasonality factors (i.e., summer, end of semester, spring recess), budgetary restrictions

on the users, and expected turnaround (response), price, and support. The final module (3.33) in this section converts the overall demand estimates into specific requests for services (i.e., statistical packages, interactive editing, etc. -- see IV.C.3), and then allocates these requests among available suppliers.

4. MARKET - The market analysis routine matches the "suppliers" and "demanders," taking into account supply constraints, scheduling priorities, communication bandwidths etc. As a result, demand for each service type may be reduced because of various capacity constraints. These calculations result in the demand matrix alluded to in paragraph III.B which describes the source and destination of all services supplied during the current time period.

5. ANALY - The analysis section provides such auxiliary computations as the determination of site turnaround and/or response times, support levels, communications load, and total workload at each site. This module also performs overall network computations in such areas as network utilization, total communications load, and tabulations of aggregate demand by service type.

6. REPORT - The report module is the last section entered in each time period. It produces reports by site, service type, user category, and for the network as a whole.

D. Implementation Conventions and Procedures

As mentioned earlier, the design and implementation of the model was carried out in a top-down structured programming manner. These techniques are well-known [8-19] and need not be expanded here. However, because of the size of the project and its relatively decentralized management, particular emphasis was placed on implementation conventions and procedures such as the following:

1. Development Library - The system was developed on an IBM 370/145 under the VM/CMS timesharing system. Each of the four people actively involved in the programming had a private account

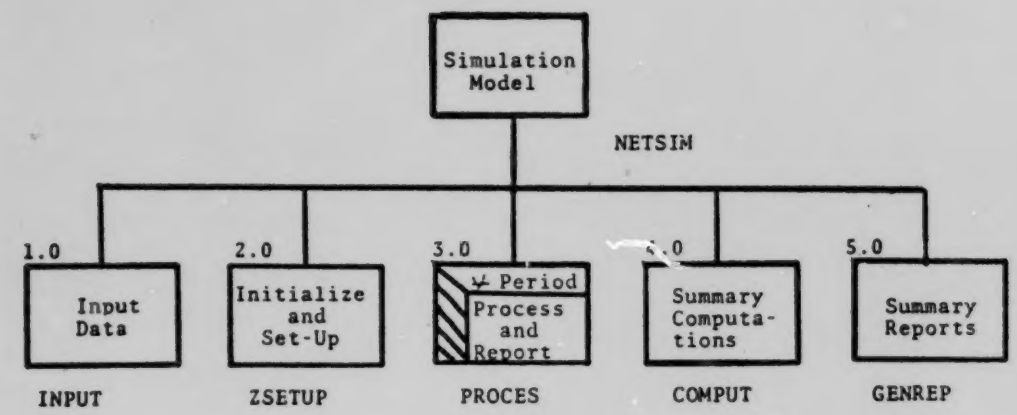


Figure III-1
Simulation Control Modules

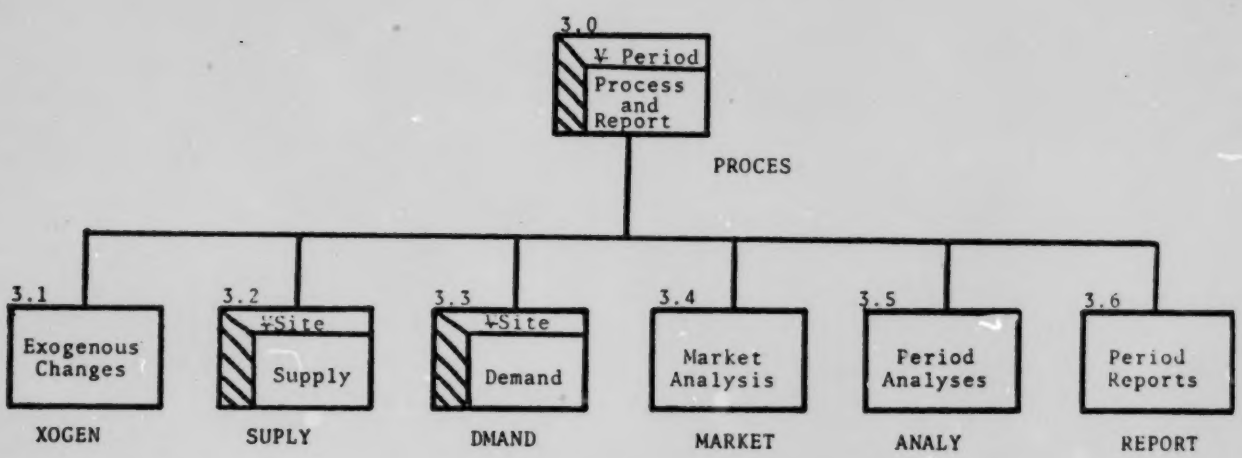


Figure III-2
Weekly Processing Sequence

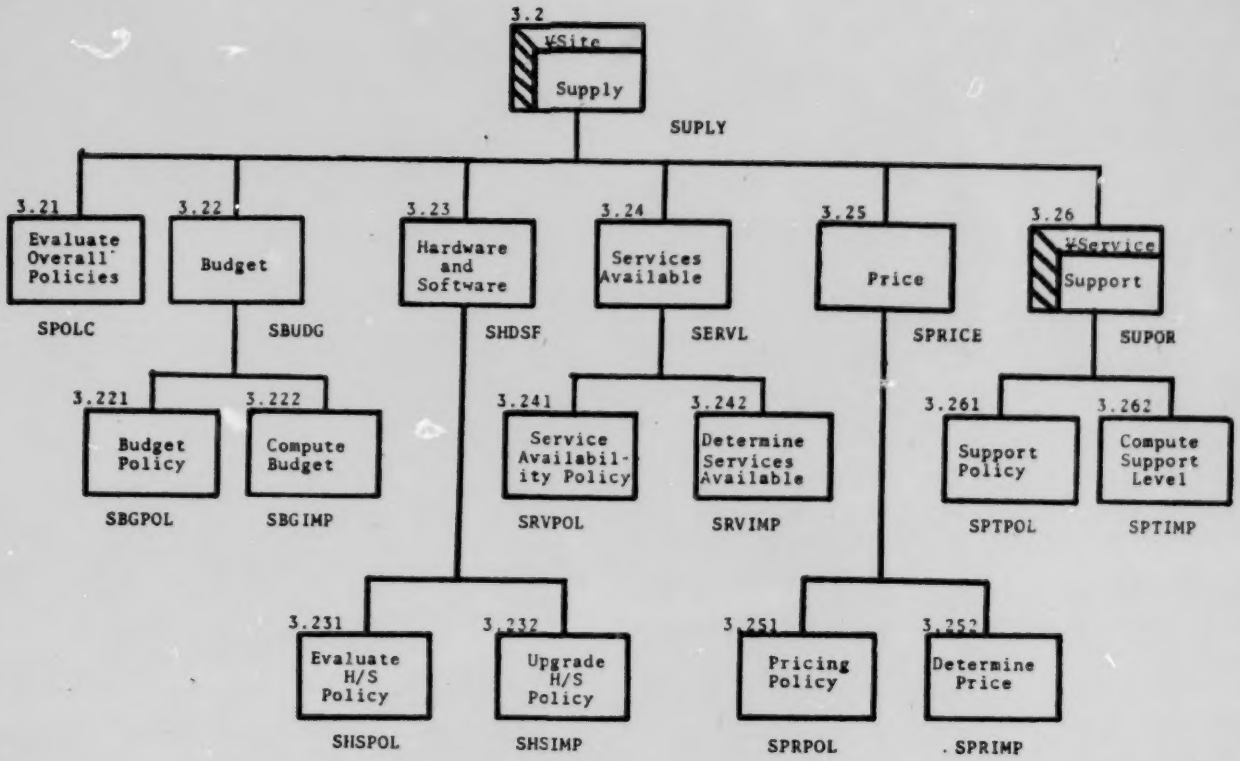


Figure III-3
Supply Determination Module

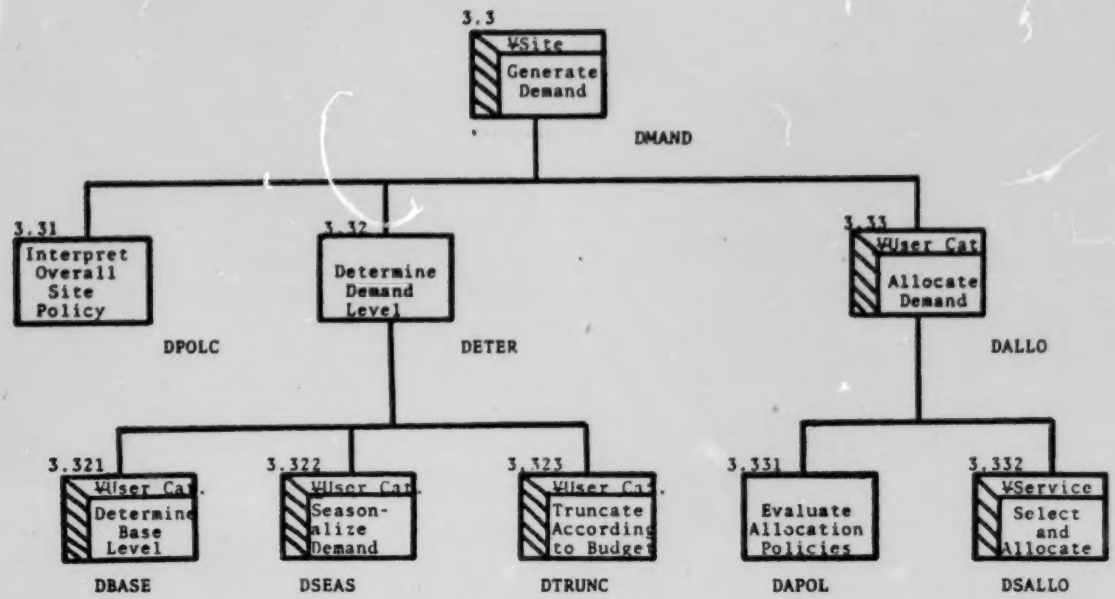


Figure III-4
Demand Estimation and Allocation Module

number. Each account number had read/write access to its own private disk space as well as to the project development library. Since CMS does not normally support multiple write access to the same library by different users, a special set of Executive routines was written to provide appropriate access and to guarantee that two users were not updating the library simultaneously. The library contains only the current working version of the simulation model.

All routines of the initial model design were originally coded as "stubs" in order to validate the flow of the model. Once the structure was validated, the stubs formed the initial system library. Using this base, all subsequent modules and modifications were developed on the programmer's individual account number and tested with the current working library model. Consequently, most library access is on a "read only" basis, with the write mode reserved for replacing existing modules with fully validated and approved versions. As added protection, the write mode is usable only with an independent set of passwords and automatic back-up and transaction logging. This guarantees that the library always contains the latest version of all validated components, and that the full model is always operational. In addition, several people can simultaneously be developing and testing new modules. Due to the simultaneous design and implementation of the model, this technique has proved to be of inestimable value in incorporating the latest design decisions into the model with a minimum of difficulty.

2. Programming Conventions - The model has been implemented using standard FORTRAN IV, since this was the most transportable of the suitable languages. It is highly desirable to enable as many institutions as possible to use the system on local computer systems.

Program coding was accomplished on-line using CRT terminals. This can sometimes cause a problem since there is no permanent record of program changes and no easy way to audit programmer effectiveness or technique. In order to maintain control of the programmers' daily interaction with the system, a feature of the

CMS system was used that allows all terminal input and output to be logged to a "console" file for later listing on a high-speed printer. This was extremely useful in the early stages of the project when there were a large number of source statements entered every day. The programmers were able to work without close supervision, while the project leaders could still review technique, help locate errors, and closely control testing and validation.

3. Common Storage Conventions - Program modules are limited to approximately 50 lines (one page) of code. All have a single entry and and single exit point. Due to the large size of the model, COMMON storage had to be used. This can sometimes be a problem in FORTRAN, since one subprogram can change a value in COMMON that may affect a seemingly unrelated subprogram. In order to minimize this possibility, several conventions were established for the use of COMMON in the model:

- Only "named COMMON" was used, thus allowing a partitioning of COMMON so that routines only saw the information they need.
- Any routine that changed values in COMMON must have had the variable(s) passed explicitly in the call statement and could not directly reference those parts of COMMON being modified.
- Copies of all COMMON blocks were kept on a separate file on the library disk. If they were changed, the new changes could be made in an automatic manner to all affected modules. This served the three-fold purpose of minimizing keypunching and data entry errors, minimizing recoding, and guaranteeing that all modules had consistent sets of COMMON blocks.

4. Flowcharts, Naming Conventions and Model Dictionary - An overall system flowchart showing the highest levels was developed early in the project. Each submodule in the system was named so that one can easily see where it fits in the overall model. For example, all routines under DMAND start with a D. In addition, each submodule was assigned a number that is keyed to both the

system flowchart and the model dictionary, Figure III-5. The model dictionary is an on-line aid used to describe all system modules. Every module is represented by module number, name, and a one-line description of purpose and function. Additional data describing input parameters and variables, internal processing, and output are entered for many modules. The dictionary therefore serves as a textual explanation of the flowcharts and an abstract of the model.

5. Coding Conventions - Each module is fully annotated within its own code, as illustrated in Figure III-6. This includes a description of function and operation, programmer name and implementation date, modification dates, tabulation of all calling and called routines, and a full set of descriptions and/or definitions of all variables used. The variables are further grouped by type -- i.e., local, common, or passed from another routine. Indication is also given as to whether or not the variable is modified within the routine. These descriptions, besides being useful for documentation purposes and imposing an organizational discipline on the programmers, were extremely helpful in program implementation and debugging.

6. Review Meetings - All-day project review meetings were held approximately monthly with the principal investigator, the project consultant, and the full project team (project manager, faculty consultant, systems analyst, and programmers). These meetings were critical for maintaining continuity of the project, since they ensured that all parties understood present status and problems, as well as future plans and schedules. All meetings included a structured walkthrough of the model conducted by the project manager and programmers, with system flowcharts and listings available as required. These walkthroughs often identified unrecognized conceptual problems that could have lead to difficulty if not resolved early. The reviews were particularly useful in the expansion of higher-level modules into more detailed lower level modules and in the specification of development tasks for outside researchers.

The review process ensured that every team member was fully conversant with all major aspects of the simulation model and related

Figure III-5
Section of Model Dictionary

```
**********************************************************************************

0.0     NETSIM    EDUCOM SIMULATION AND GAMING MODEL,  AS OF  7/15/76

**********************************************************************************

1.0     INPUT     READ INITIAL DATA

   1.1      IRNCTL    RUN CONTROL PARAMETERS

      1.11     INTAC     INTERACTIVE DATA
      1.11     1 INPUT   NONE.
      1.11     2 PROCESS ACCEPT DATA FROM TERMINAL (UNIT 5)
      1.11     3 OUTPUT  NUMBER OF PERIODS (NN).
      1.11     3 OUTPUT  DATE OF RUN (MM,MD,MY)
      1.11     3 OUTPUT  RESTART INDICATOR (IRSTRT)
      1.11     3 OUTPUT  RUN TIME COMMENTS (FILE MODOUT)
      1.12     IRLGIN    RESTART CONTROL MODULE

   1.2      INETWK    NETWORK DESCRIPTIVE DATA

      1.21     ISYSPR    SYSTEM PARAMETERS
      1.21     1 INPUT   NONE.
      1.21     2 PROCESS READ FROM FILE 'MPARMS' (UNIT 3)
      1.21     3 OUTPUT  NUMBER OF SITES (NSITES)
      1.21     3 OUTPUT  NUMBER OF SERVICE TYPES (NTYPES)
      1.21     3 OUTPUT  NUMBER OF RESOURCES (NRES)
      1.21     3 OUTPUT  NUMBER OF OVERALL POLICY SEGMENTS (NOPOLC)
      1.21     3 OUTPUT  IN-HOUSE RATING IMPROVEMENT FACTOR (DEBUMP)
      1.21     3 OUTPUT  SMOOTHING CONSTANT (ALPHA)
      1.21     3 OUTPUT  NETWORK COMMUNICATIONS CHARGE (COMCST)
      1.22     INETPR    NETWORK PARAMETERS
      1.22     1 INPUT   NUMBER OF SERVICE TYPES (NTYPES)
      1.22     2 PROCESS READ FROM FILE 'MORDER' (UNIT 1)
      1.22     3 OUTPUT  COMMUNICATIONS MAP (COMO)
      1.22     3 OUTPUT  PRICE DISCOUNTS (DSCNT)
      1.22     3 OUTPUT  TURNAROUND MODULE TABLE (ITAB)

   1.3      ISIDAT    SITE DATA AND PARAMETERS (FOR EVERY SITE)

      1.31     ISINFO    GENERAL SITE INFORMATION.
      1.31     1 INPUT   SITE NUMBER (ISITE)
      1.31     2 PROCESS READ FROM FILE 'MSITE' (UNIT 4)
      1.31     3 OUTPUT  OVERALL SITE POLICY (IOPOLC),
      1.31     3 OUTPUT  REPORT SELECTION INFORMATION (ISRP,IIRP,ICRP).
      1.32     ISUPPL    INITIAL SUPPLY FACTORS  FOR EACH SERVICE
      1.32     1 INPUT   SITE NUMBER (ISITE)
      1.32     2 PROCESS READ FROM FILE 'MSUPPL' (UNIT 11)
      1.32     3 OUTPUT  BUDGET INFORMATION (BUDGET)
      1.32     3 OUTPUT  RELIABILITY (RELY)
      1.32     3 OUTPUT  HOURS 'UP' (HRSUP)
      1.32     3 OUTPUT  SERVICES OFFERED VECTOR (SRVAVL)
      1.32     3 OUTPUT  TOTAL CAPACITY FOR EACH RESOURCE  (TOTCAP)
      1.32     3 OUTPUT  RESOURCE MAPS FOR EACH SERVICE (RIFMAP)
      1.32     3 OUTPUT  AVERAGE SERVICE RATE AND PRIORITIES (AVGRAT,SRVPRI)
      1.32     3 OUTPUT  PER UNIT PRICE FOR EACH RESOURCE (PRIRES),
      1.33     IDEMAN    INITIAL DEMAND FOR EACH USER CATEGORY
      1.33     1 INPUT   SITE NUMBER (ISITE)
      1.33     2 PROCESS READ FROM FILE 'MDEMAN' (UNIT 2)
```

Figure III-6
Sample Program Listing
(Segment)

```
      SUBROUTINE PRICAL(ISITE,KTYPE,PRICE,PRIRES)                    PRI00010
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPRI00020
C                    PRICAL                                         C PRI00030
C  NETWORK SIMULATION MODEL - PRICE ROUTINE                         C PRI00040
C                                                                   C PRI00050
C  THIS SUBROUTINE CALCULATES PRICES FOR THE                        C PRI00060
C  SPECIFIED SITE BY SERVICE TYPE.                                  C PRI00070
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPRI00080
C  CREATED: 2/5/76                                                  C PRI00090
C  PROGRAMMED BY: STEVE BENSINGER                                   C PRI00100
C  LAST MODIFIED: 5/18/76 JOE PUGLISI                               C PRI00110
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCPRI00120
C                                                                   PRI00130
C  LOCAL VARIABLES:                                                 PRI00140
C       IRES     SYSTEM RESOURCE NUMBER                            PRI00150
C       TEMP     TEMPORARY VARIABLE                                PRI00160
C                                                                   PRI00170
C  COMMON VARIABLES:                                                PRI00180
C                                                                   PRI00190
      COMMON /SYSPM/NSITES,NTYPES,NRES,NCATS,NOPOLS,NCRPTS,NSRPTS,NKRPTSPRI00200
C                                                                   PRI00210
C$$   NSITES  NUMBER OF SITES ON THE NETWORK                        PRI00220
C$$   NTYPES  NUMBER OF SERVICES ON THE NETWORK                     PRI00230
C$$   NRES    NUMBER OF SYSTEM RESOURCES                           PRI00240
C$$   NCATS   NUMBER OF INCOME/EXPENSE CATEGORIES                  PRI00250
C$$   NOPOLS  NUMBER OF OVERALL POLICY SEGMENTS                    PRI00260
C$$   NCRPTS  NUMBER OF CASH REPORTS                               PRI00270
C$$   NSRPTS  NUMBER OF SPECIAL REPORTS                            PRI00280
C$$   NKRPTS  NUMBER OF SERVICE SPECIFIC REPORTS                   PRI00290
C                                                                   PRI00300
C                                                                   PRI00310
C  PARAMETERS:                                                      PRI00320
C                                                                   PRI00330
C$$N  ISITE   SITE NUMBER                                          PRI00340
C$$N  KTYPE   SERVICE TYPE (CODE)                                  PRI00350
C                                                                   PRI00360
      DIMENSION PRICE(20,10),PRIRES(20,10)                          PRI00370
C                                                                   PRI00380
C$$Y  PRICE   PRICE AT JSITE FOR KTYPE                             PRI00390
C$$N  PRIRES  PRICE AT JSITE FOR RESOURCE IRES                     PRI00400
C                                                                   PRI00410
CXX CALLS:  NONE                                                    PRI00420
C                                                                   PRI00430
C  CALLED BY: ZCOMP                                                 PRI00440
C                                                                   PRI00450
C     WRITE(6,1)                                                    PRI00460
1     FORMAT(1X,'ENTERED PRICAL')                                   PRI00470
C                                                                   PRI00480
C.....CALCULATE PRICE.....                                          PRI00490
C                                                                   PRI00500
      TEMP=0.                                                       PRI00510
      DO 100 IRES=1,9                                               PRI00520
         CALL RIFMAP(ISITE,KTYPE,IRES,X)                            PRI00530
         TEMP=TEMP+(X*PRIRES(ISITE,IRES))                           PRI00540
100   CONTINUE                                                      PRI00550
C                                                                   PRI00560
C.....ADJUST PRICE FOR PRIORITY AND STORE.....                     PRI00570
C                                                                   PRI00580
      CALL RIFMAP(ISITE,KTYPE,10,X)                                 PRI00590
      PRICE(ISITE,KTYPE)=TEMP*X                                     PRI00600
C                                                                   PRI00610
```

research activities, as well as the needs and desires of the eventual model users. It also served to ensure compatibility between the output of outside researchers and the model requirements. Hence, steady progress on system implementation took place in parallel with work on unresolved design and implementation issues. In addition to the monthly review meetings, more traditional detailed discussions were held at more frequent intervals. These did not include the entire team and usually focused on code rather than design and logic issues.

E. Simulation Run Requirements

All of the necessary run commands have been automated in CMS Executive procedures. Thus, the entire system is executed by simply entering "SIMRUN" at the main console. This procedure must be modified if the model is run in a different computer environment than the one in which it was developed. Many file definitions and load and start procedures must be specified. These are described in greater detail in Appendix III - Model User's Guide.

The simulation model depends on a fairly large data base. To use the model, the data must be collected, assembled, properly formatted, and incorporated into a set of on-line files. The actual execution of the simulation model is far simpler than the proper definition of the data files upon which it depends.

The size of the model is about 15,000 lines of source code. Currently the model requires 600,000 bytes of main memory, operating in a virtual memory environment. Each site currently requires approximately 300 records of 80 characters each, although this could be reduced by data compression.

1. Network Data - The first step necessary for use of the model is a definition of the network being modelled for a given simulation run. Parameters such as the number of sites, services, and time periods, are entered interactively at the start of each run. Consistent with these entries, stored files must be available

to describe the relationship between sites with respect to both communications costs, charges, and pricing structures. Any discounts or surcharges between sites must also be included.

2. Site Data - Once the network is defined, the system must be able to access a large amount of pre-stored data for each of the sites included in the network being modelled. The items that must be specified for every site includes the site's categorization of computer users; the amount and type of services each category uses; the budget for these users and for the computer center; the restrictions on each category of users as to where jobs may be run; and each category's sensitivity to prices, turnaround, and support. The seasonality and growth of demand at each site is also required. Sites that are suppliers of services must also describe the services available, the impact of these services on the site's computer resources, and the cost, type, and amount of additional capacity that could be obtained.

Finally, policies or practices at each site must have been defined. Where the stored menu of standard policies is not adequate, new ones must be written and inserted. The policies are explained in Appendix I and methods for adding or modifying existing policies are described in Appendix V (Modifications Guide).

3. Other Files - A number of tables (Appendix IV.H) are required to describe the sites being simulated. These tables provide titles and text for output reports and include site names, service type names, and computer resource names.

4. Output - Output from the model may be directed to an on-line terminal, disk files, or a high-speed printer. Available outputs (Appendix III.B) include model reports, trace information (used for debugging), and a log file. The log file is essentially a periodic dump of those site values considered critical or interesting from an experimental standpoint. This file may be written on magnetic tape for later off-line analyses.

F. Model Documentation

It is of particular importance that this model be usable by various institutions and capable of being modified by an institution to meet unique needs not included in the general model. Consequently, documentation must be available for early use of the model by participating institutions during Phase III of the project. If these goals are to be attained, adequate documentation has to be available throughout the life of the project. Hence, special attention was given to maintaining model documentation on a current basis. In addition to the usual textual and algorithmic descriptions, emphasis has been given to several techniques as outlined below. Current versions of all documents mentioned are included in the Appendices to this report.

1. System Diagrams - The basic structured diagrams described earlier were completed during the design phase of the project. These diagrams include all major system modules and illustrate full system flow. Although occasional minor modifications occurred in the higher levels of the diagrams throughout model implementation, this aspect of the documentation was essentially complete before any significant coding was started. It should be emphasized, however, that the lower modules are in a continuous state of expansion. Thus, the model will never be "complete," and evolution can continue indifinitely without affecting previous documentation or implementation.

2. Module Dictionary - The on-line dictionary, Figure III-5, also was maintained on a current basis. Routines have been written to reorder or to extract modules in various combinations, allowing the on-line listing of detailed descriptions of any or all model segments.

3. Internal Documentation - Each module is fully annotated within its own code as described earlier. This ensures that the documentation is thorough and up-to-date and greatly simplifies the preparation of external documentation.

4. HIPO - HIPO (Hierarchical, Input, Processing, Output) diagrams have been maintained for those modules not adequately described by the system diagrams. A sample is illustrated in Figure III-7. This documentation technique is well known and will not be discussed here[20].

5. Help Files - On-line documentation for execution of the model and related functions was also maintained in the form of "help" files. These files contain descriptions of the form, function, parameters, and usage of all procedures required for the model execution. These files are contained in Appendix IV.

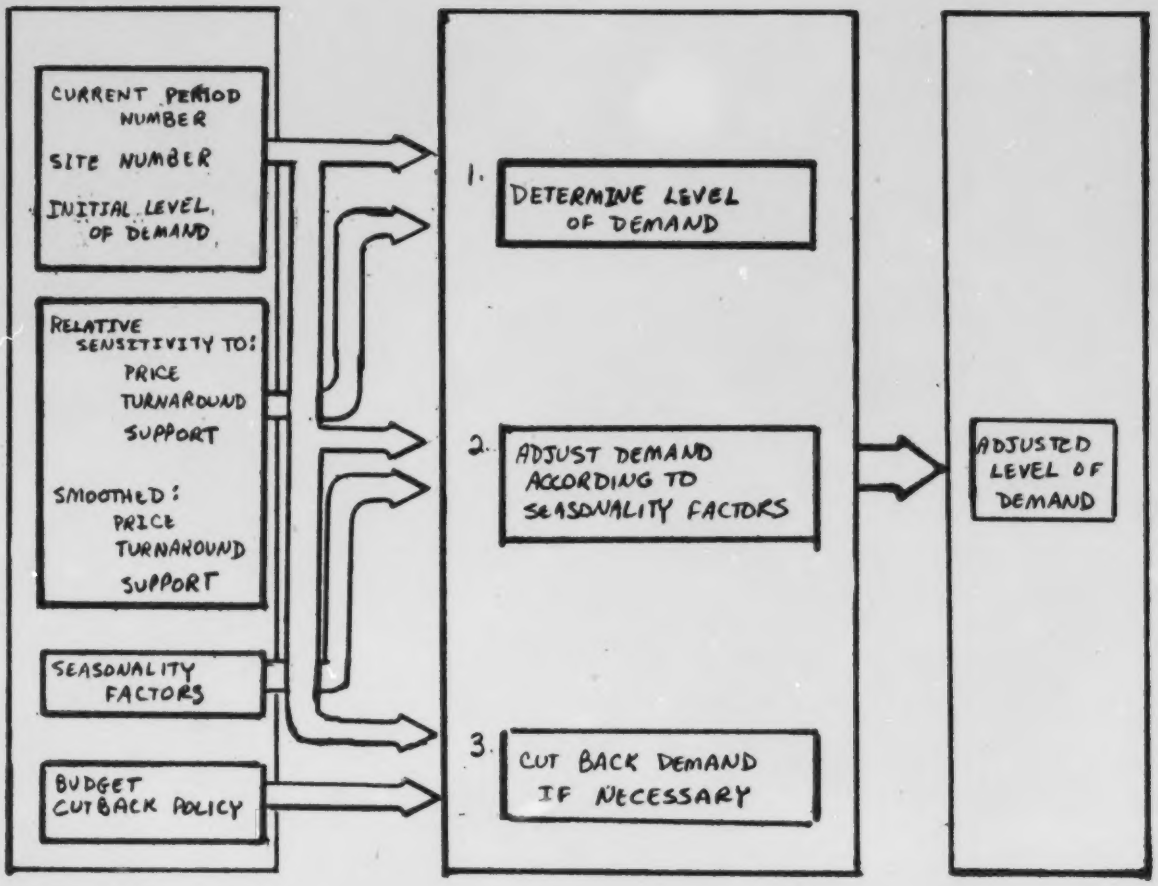
6. User's Guide - The user's guide, Appendix III, describes the procedures for running the model. The model depends on a large amount of input data, the nature and form of which are described in this section. The guide also includes instructions for defining runs, creating files, responding to interactive requests, and obtaining optional outputs. This document will be particularly useful in using the model for experiments.

7. Reference Guide - The model reference guide, Appendix IV, includes detailed documentation on all of the above. In addition, all major variables, programming conventions, system environment, files, utilities, and Executive (EXEC) procedures are described.

8. Modifications Guide - Several model areas in the model are likely to remain the subject of frequent modification. This is particularly true for policy modules and report generators which may be altered or added for specific experiments. This guide (Appendix V) is primarily aimed at simplifying and standardizing the procedures for performing these types of modifications.

Figure III-7
Sample HIPO Diagram

DIAGRAM NUMBER: 3.32 NAME: DETER DESCRIPTION: ESTIMATE DEMAND BASED ON GROWTH, SENSITIVITY, SEASONALITY, AND CUTBACK POLICIES.



NOTES	ROUTINE	REFERENCE
1. ESTIMATE DEMAND BASED ON EXPECTED GROWTH AND PAST TURNAROUND, PRICE, AND SUPPORT LEVELS	DBASE	3.321
2. ADJUST DEMAND BY A SEASONALITY FACTOR FOR THE CURRENT TIME PERIOD	DSEAS	3.322
3. IF A BUDGET CUTBACK POLICY IS IN EFFECT, CHECK TO SEE IF DEMAND MUST BE REDUCED	DTRUNC	3.323

G. Validation

1. Design Validation - In a large, complex system of this type, one of the major problems usually lies in simply validating that the model "does what it is supposed to do," and functions according to specifications. Use of the top down approach described earlier permitted a continuous monitoring of the model from the earliest stages throughout the implementation effort and helped to minimize this problem. For example, at the end of project week two it was confirmed that module NETSIM called all top-level modules correctly, and that all variables and parameters were properly passed between routines. Hence, it was possible to "sign-off" interactions between modules 1.0 through 5.0, and treat the expansion of each of these modules independently.

By continuing the above process on a hierarchical basis, it is possible to state with reasonable confidence that the model functions as designed. Although there may be problems in specific algorithms, these are generally in the lowest level modules which is also the level at which continuous expansion is taking place. However, such problems are easily isolated and corrected and will not affect overall system function and design.

2. Implementation Validation - One of the difficulties in validating this system is that the simulation is modelling an environment that does not currently exist. Once the model had passed simple plausibility tests, it was clear that more refined validation called for a controlled environment in which results were already known. To accomplish this, the site with the most comprehensive (and reasonable) data was chosen as the test vehicle and used for initial testing. The model was run with a one-site network, and the results compared to the data supplied by the test site. This immediately uncovered a number of discrepancies, such as the estimates of throughput and turnaround, that were traced to both programming errors and data errors. Once the one-site model was fully debugged, the site was replicated several times in order to run a multi-site model with identical sites. The objective here was to ensure that the model did not

introduce any spurious network flows.

A good part of the model, however, is only applicable to an environment in which there are network flows. In order to validate the areas of the model involved with network flows, the multiple identical site configuration was perturbed in a number of ways.

The first perturbation was to change the data of one site in such a manner that it would be forced to use the network to accommodate its demand for computational facilities. To accomplish this, the computing capacity at that site was reduced to a fraction of its original value, and its hardware budget reduced by a corresponding amount (thus freeing up budget dollars to be spent outside). The demand at that site was kept at the original level. Concomitantly, another site's capacity was increased to ensure that the first site had a place to go.

Similar runs were made with reduced prices at two sites to see if flow of work would gravitate to a site with extremely low prices. Other runs were made with similar perturbations to turnaround and level of support. The final model validation runs were done with each of the sites configured to exhibit "prototypical" behavior of what was thought might be possible site behavior patterns. For example, policies for one site were chosen to exhibit cost-conscious behavior, while another site was defined as an entrepreneurial center. A mixture of capacities and services offered at the different sites was introduced. The model was run with these data to ensure that the different policies made the sites behave in the expected manner. At this point the model was ready for testing with site data as provided by the participating institutions.

3. Data Validation - There were a number of problems with data validation. The primary problem was consistency. Data were frequently found to be inconsistent with external data and/or with other data in the model. It was found, for example, that sites had reported daily data where the model expected weekly. Problems of this type were relatively easy to spot. A number of more basic

errors were found in such performance data as the average CPU time of the different service types and the total number of jobs or connect hours per week. When these values were used in the model, anomalies were often discovered, such as estimated CPU use that exceeded the total capacity available. These errors were detected by running each site alone in the model to verify that all the capacity utilizations were reasonable and that the correct number of jobs were being run in each of the service types.

The hardest data validation problem was in the lack of consistency in the definition of service types. Each site was asked to give its estimate of the number of jobs of each service type run per week, and what the impact of each job was on that site's resources. Unfortunately, a "small FORTRAN" job on IBM 370/135 may not be the same as a "small FORTRAN" job on a CDC 7600. In order to discover discrepancies of this type, a program was written to tabulate and to print the resource impact factors for all sites for a particular service type. By comparing this to benchmark data for a small set of service types, it was then possible to discover and resolve many of the definitional inconsistencies between sites.

4. Future Validation Requirements - Needless to say, in a developing model such as this one, validation is a reoccurring and non-trivial concern. For this reason the validation process can never be called complete. It should also be recognized that verification that the model performs "correctly" as described above is the easy part of the validation process. More difficult is verification that the model is a useful representation of the real world. I.e., is it simulating the right things, and if so, does it operate at the right level of aggregation? Is it believable; can decisions be made based on the model results, etc.? This deeper level of validation is not yet possible with the existing model. It will begin in Phase II when the model is enriched with estimates of the actual decisions and policies of the participating institutions.

IV. RESULTS OF BACKGROUND STUDIES

A. Purpose and Approach

In parallel with the design and implementation of the simulation model as described in Section III several different areas were identified in which algorithms, procedures, or new representational concepts were needed. The results of these efforts form the theoretical base for many segments of the model. It should be noted that these studies were all undertaken to fill specific needs of the model and/or the project. Although virtually all of these topics represent areas where there is a great need for basic research, in-depth efforts were generally beyond the scope of this project. More typically, emphasis was on doing "what had to be done" in order to obtain reasonable representations.

The approaches varied over a wide spectrum. A number of studies (perceptions of computer services, user categorizations, user services and support) performed by the project staff consisted of continuously evolving a representational framework until a point was reached that satisfied both outside reviewers and model requirements. Other studies (i.e., service type definition, benchmark testing) required that vast amounts of empirical data be collected and tabulated. In some cases (network organization alternatives, user services and support, and pricing) informal discussions were held with recognized experts and some relatively subjective "consensus" opinions were reached.

In two of the most critical areas (performance modeling and estimation, and workload representation), outside experts were available who had done substantial amounts of relevant research, and it was only necessary to support the extra work required to adapt the proven technologies to the model requirements.

This chapter summarizes the results of many of these efforts. The variation of format, sophistication of research, and depth of

analysis reflect the differences in model requirements; and hence in the definition and pursuit of the individual studies.

B. Network Organization and Administration

Although technical issues are still important factors in establishing a computing network, the dominant impediments to sharing are very likely to be non-technical in nature. Accordingly, a major concern of the study was an examination of a wide range of organizational and administrative alternatives that might influence the behavior of the network.

It was necessary to identify these issues so that the model might be brought to bear on their analysis on many levels during all phases of the project. These may be examined through use of the model in a variety of ways. Some are clearly policy decisions that may be studied by a proper combination of policies in the appropriate areas of the model. Others represent constraints or a range of possible considerations which may be examined by proper specification of model parameters or input data. Typical issues of direct concern are the following:

- Network administration -- including consideration of the location of control over accounts, billing, resource use, types of services offered, administrative conventions.
- Centralization -- including consideration of centralization of general capacity, the centralization of specialized capacity or services, the relation between economies of scale and economies of specialization, and the relation between economies and diseconomies of scale for various types of services
- Methods of charging for centralized facilitating services -- (e.g., billing, user services, etc.), including fixed monthly fee, unbundled charges for each service rendered, and charges levied against suppliers.
- Control over prices by a central network organization -- ranging from no control at all to detailed price regulation. Includes consideration of subsidies, restrictions on "unfair" price competition, and standardization of prices.

- Alternative pricing arrangements -- including "spot" versus long-term contracting, fixed monthly charges versus charging for each service rendered, and differential pricing (e.g., priority level, peak versus off-peak, etc.).

- Mechanisms for billing and reporting -- ranging from bilateral arrangements between each buyer-seller pair, to multilateral arrangements through a central network organization.

- Alternative budgeting arrangements at user institutions -- ranging from centralized budgeting of the computer center (with allocation of capacity through non-discretionary "computing dollars"), to revenue generation for the computer center through a cost recovery mechanism in which users are charged "real" money.

- Capacity adjustment -- including consideration of the ability and willingness of institutions to reduce (expand) local capacity in response to an increasing net import (export) of services.

- Resource migration -- including consideration of the possibility that the "resource rich" might become richer while the "resource poor" might become poorer.

- Service guarantees -- including consideration of supplier guarantees as to price, quality, and quantity of services provided and purchaser guarantees as to the quantity and timing of purchases.

- Mechanisms for providing user services -- (e.g., written documentation, on-line directories, consultants, etc.), ranging from direct service from the supplier, to service from a local "distributor."

- Software development -- including consideration of development incentives, royalties, and support of development activities

After reviewing issues such as these, it was decided that several approaches in combination would be needed to encompass the wide variety of possible network organizations and administrations. As a result there are currently two basic methods of representing these in the model. One is through the menu of policies available to represent various attitudes and decision-making behavior at individual sites. The other is through network parameters which may be selected for the network as a whole. Policies are discussed

in detail in Appendix II and the parameters are reviewed in Appendix III of Volume II. Most of the work during Phase I has focused on making the model adaptable and rich enough to accommodate a wide range of organizational and administrative alternatives. Future project phases will therefore be able to examine these issues in more detail.

C. Representational Concepts

If the simulation model is to portray a possible "real" network, the structural elements of the model must be reasonable approximations of their real life counterparts. Consequently, such areas as user perceptions of computer services, institutional categorization of users, and descriptions of workloads at various levels were the topics for specific studies. The representations formulated provided the building blocks for the development of the model.

1. Perceptions of Computer Services - Within an organization, there are usually several levels of perception of computer services and workload. On one hand, there are administrators who have budget and policy-making responsibility, but whose knowledge of computing may be quite limited. At the other extreme, there are computer center operations people who perceive their role as suppliers of raw capacity in terms of CPU cycles, bits of main memory, characters of input/output, etc. Each of the existing levels at any site may be represented. Currently, the model contains four such levels under the general categories of: Administrator, User and/or Supplier, Computer Center Director, and Performance Analyst. These different levels are necessary because of the rather dissimilar viewpoints of the individuals involved at each level.

a. Administrator Level - Administrators, defined as those individuals with organizational policy and resource commitment responsibility, are likely to view computing in terms of internal budget lines within the organization. Thus, one university may differentiate between student jobs, faculty research, and administrative data

processing; while another might view usage and formulate computer budgets by school (i.e., Engineering, Arts and Science, Business, etc.). In general, each site has its own viewpoint and its own set of user categories.

b. User/Supplier Level - Users and suppliers of services think in terms of the type of processing required. A user may have a large FORTRAN program to run, or perhaps a need for an interactive graphics package. Individual users select supplier sites based on such variables as software availability, turnaround, price, and support. Installations, therefore must describe available services in the jargon familiar to the individuals who will be using those services. Note that at most sites it is this level that is closest to the network "standard" service type. Hence, translations from site categorizations to network service types are done at this level and based on the installation's description of available services (services supplied).

c. Computer Center Director Level - While the computer center director is interested in, and responsive to, the above levels, he needs further information in order to make decisions relative to configuration, staffing, system software, and other required resources. Information such as CPU usage, lines printed, cards read, file space required, memory usage, and number of tape mounts is needed. Thus, incoming work at a site has to be presented in a way that a computer center director can obtain this hardware loading data. For him, each service type is described in terms of appropriate resource requirements. Total system loading can then be estimated by summing the per unit resource requirements of all jobs in the workload.

d. Performance Analyst - Once the total workload at a site has been defined, the task still remains to estimate site performance as a function of that workload. Cyclic queueing models and related techniques are currently being developed for this purpose (see Section IV-D.2). In general, these techniques require that the workload be described in terms of resource utilization. This information is a more detailed version of that used by the computing center director described above.

Within the model it is necessary to represent each of these perceptions and to translate from one to another. In this way decisions made from one point of view can be expressed in ways meaningful to the others. Currently the administrative level

determines the policy selection for a particular site. This is typically done in terms of user categories (or budget lines). The policies then determine the methods for handling service types at the user/supplier level and raw resources at the computer center director level. These concepts are explained more fully in the following sections.

2. User Categorizations - As discussed earlier, the highest level of perception in the model is at the administrative level. This is where major policies and budget constraints originate and are controlled. Note that administrators do not deal with specific computer services or resources, but rather with broad categories of users.

Each site has its own site-specific user categories which represent logical internal administrative divisions. In asking sites to specify their user categories, two guidelines were provided:

a. Each category must represent an identifiable budget line or funding source.

b. Each category must be relatively homogeneous with respect to rules, policies, and constraints on computer usage. For example, rules for "going outside" would be consistent within any one user category.

Most universities presented categories such as: student instruction, externally funded research, internally funded research, administration, external users, and computation center systems staff. A research institute, on the other hand, had only two, internal and external users. A few of the universities grouped their users along organizational lines, (e.g., College of Engineering, Law School, Medical School, etc.).

A major implication of the above categorizations is that each user category develops its demand for computer services independently of others at that site. Some users (e.g., administrative

computing) may be relatively insensitive to price and turnaround, for example, while other users at that site (e.g., instructional users) may be very price and turnaround sensitive. In any case, it is necessary within the model to:

a. Permit each site to define its own unique user categorizations.

b. Maintain separate budget and policy structures for each user category.

c. Provide separate estimates of aggregate demand for computational services for each category.

d. Provide a means to translate (map) aggregate demand for each category into service-specific demand.

e. Allocate this service-specific demand among available sources of supply by the application of criteria which may differ for each user category.

These user categories will play an even larger role in Phase II of the project, when the actual policies and procedures of the participating institutions are incorporated into the model. As the policies, rules, and restrictions on these categories are developed, they will evolve into an accurate representation of the specific institutions to which they belong.

3. Service Types and Workload Representation

a. Problem - One of the most difficult conceptual tasks in this project was the definition of work (or service). Prospective network users, even those few that have a good idea of what they would like to be doing, are likely to have requirements whose characteristics are very different than the present job stream at the desired supplier site. Further, their perception of what constitutes a

unit of work, and what that is worth, will often be incompatible with the viewpoint of the supplier. For example, some of the participating institutions in this project describe their workload using jargon such as: FORTRAN, COBOL, GPSS, STATPACK and the like; others talk only at the user category level (student jobs, administrative data processing, and faculty research); and some prefer terms like compute-bound, I/O bound, large batch, and heavy on-line usage. Furthermore, "similar" programs and services available at multiple sites are not always compatible or transferable -- especially when dissimilar host computers are involved.

b. Approach - Although, as mentioned above, little standardization exists among individual sites as to how work (jobs, services) is described, it was decided to try to develop a consistent work description for network purposes (though none of the individual sites need use that particular description). The initial goal was to define a set of service types, limited if possible to less than 50 in number, such that the workload of each network member could be adequately approximated by an enumeration of the numbers of jobs in each category per unit time (week). Definitions of service types should specify domains or ranges for a number of site-independent parameters such that any job, regardless of origin, could be assigned a category designation that would permit an adequate determination of which sites might be appropriate for its processing and of its processing characteristics at those sites.

Other desired characteristics of such a classification included:

- All categories should be meaningful to users, although they need not match current in-house groupings. Where differences exist, a transformation (conversion) must be feasible.
- The network categories should be expressible in a form that is appropriate for use in performance analysis and prediction at individual sites. I.e., jobs will have known resource requirements and attributes (CPU seconds, memory words, etc.).
- All items within a given classification category must be relatively homogeneous in terms of machine impact at any given site.

c. Service Type Dimensionality - Early attempts to organize the categorization process yielded a minimum of four factors, or dimensions, for each category definition.

i. Job Type (qualitative), e.g., FORTRAN compilation and execution, on-line data entry, execution of statistical package, etc.

ii. Resources Required (quantitative), e.g., processor memory, card reader, disk I/O, printer, etc.

iii. Running Time (quantitative), e.g., short, medium, long.

iv. Priority (quantitative).

It was immediately evident that any four-dimension matrix would quickly grow far beyond the 30 to 50 element size limitation imposed by the then-current simulation design. Accordingly it was decided that, in order to stay within the size limitation, the service type definitions should be based primarily on Job Type (a one-dimensional list), with multiple entries for some job types

to accommodate suitable modifiers relating to running time and priority. Resources required would be implicit in the job type and would not need a separate dimension.

The complete list of service types was developed in qualitative but final terms, and each institution was asked to judge which types best represented the actual job characteristics at their institution, to specify the resources required for an average job in each service type, and to estimate the number of "average" jobs of the service type which would adequately represent the total level for that service type.

The basic assumption underlying this approach is that service types need not be quantitatively comparable between institutions. For modeling of the internal use of an institution's computer services, which probably would still constitute the bulk of the services even after national networks are available, this assumption is entirely consistent. Between institutions, it is assumed that the characteristics of a job will change depending on the institution at which it is run.

For example, an ABC job of a given service type, if run at XYZ University, would take on the resource requirements for jobs of this service type as defined by XYZ rather than as defined by ABC. In other words, ABC users at XYZ will tend to behave like XYZ users rather than ABC users. Since demand is allocated by policy and doesn't change rapidly, this assumption appears to be more reasonable than the converse assumption that a job originating at a specified institution will have the characteristics of the jobs of that service type for the origination institution, independent of where the job is actually processed.

d. Initial Categorization - The initial list of service type descriptions was a common-sense development of the list obtained from institution responses to Questionnaire I. Service types known to exist,

but not mentioned by the institutions in their responses to the questionnaire, were added. Several service types mentioned in the replies, such as little-used compilers, were lumped into catch-all categories labelled "other." A number of service types were repeated, with modifiers such as "short," "medium," "long," "low CPU utilization," and "high CPU utilization."

This initial tentative list of service type descriptors had 42 entries in three general classes, "Batch, with very restricted resource allocation," "Batch, general," and "Interactive." It was assumed that all interactive jobs would run under the highest priority, assigned the number 1, and that the restricted batch jobs would all run under a number 2 priority. In the second questionnaire, users were asked to estimate resource usage as a function of priority (four levels) for each service type in the class of general batch jobs. There were 20 entries in the general batch classification, which, with priority modifiers, represented 80 service types, so the total list had 102 entries. This initial list appears in Questionnaire II (Appendix VII).

e. Final Categorization - As resource usage data were compiled from replies to the second questionnaire, entries in the service-type list were combined again to reduce their number in accordance with storage limitations within the simulation model. Little-used service types were merged with similar categories, and differentiations that were difficult for individual sites were eliminated. All members of the restricted resource allocation class of batch jobs, for example, were lumped into a single "Fast Batch" service type. Similarly, all of the "compile and bomb or short run" jobs in the general batch classification were lumped into a

single "Debug Runs" service type and assigned a number 3 priority. Among the interactive jobs, the high and low CPU utilizations were combined, eliminating that distinction. The rest of the consolidation process consisted of lumping together two, three, or, in some cases, all four priority levels and assigning a single priority number. The final list of service type names has 44 entries, distributed as follows:

- 11 Interactive job types with #1 priority.
- 1 Fast Batch entry with #2 priority.
- 32 General Batch job types with priorities ranging from #3 to #6.

Four additional service type entries were reserved for special or unique services that a facility might offer. The final list appears as Figure IV-1.

f. Status - The above characterization is useful in several ways. Providers of computing services can describe their offerings to remote users in terms of the network standard list. Similarly, once prospective buyers express their needs in this form, they can easily investigate the availability of desired services on the network. All network flows (workflows between sites) are now expressed using this common denominator.

It is recognized that no such list is likely to exactly match the services offered at any single site. Further, the standard definitions for job types may not be consistent with those at individual sites. However, the nature of most inconsistencies is in degree (e.g., size of FORTRAN jobs, average

Figure IV-1
Service Types

Service Type	Priority
1. Fast Restricted Batch	2
2. Debugging Runs	3
3. FORTRAN Program Development	3
4. FORTRAN Program Development	4
5. FORTRAN Program Development	5
6. COBOL Program Development	3
7. COBOL Program Development	4
8. COBOL Program Development	5
9. PL/1 Program Development	4
10. PL/1 Program Development	5
11. Assembler Program Development	3
12. Assembler Program Development	4
13. Other Program Development	3
14. Other Program Development	4
15. Other Program Development	5
16. Graphics Packages	3
17. Problem Oriented Packages	3
18. Problem Oriented Packages	6
19. Short Statistical Packages	4
20. Medium Statistical Packages	5
21. Long Statistical Packages	6
22. Short Number Crunching	3
23. Short Number Crunching	4
24. Short Number Crunching	5
25. Medium Number Crunching	5
26. Medium Number Crunching	6
27. Long Number Crunching	6
28. Short File Manipulation	4
29. Short File Manipulation	5
30. Medium File Manipulation	5
31. Medium File Manipulation	6
32. Long File Manipulation	5
33. Long File Manipulation	6
34. Data Access, Read Only	1
35. Data Entry	1
36. Low Activity Text Editing	1
37. Intensive Text Editing	1
38. Terminal BASIC	1
39. Terminal FORTRAN	1
40. Terminal PL/1	1
41. APL	1
42. Other Terminal Languages	1
43. CAI	1
44. Interactive Problem Oriented Packages	1

connect minutes per interactive session, number of different sizes of statistical packages that must be specified in order to get meaningful discrimination), rather than concept.

Data describing present workloads in terms of the service types shown in Figure IV-1 are now available from most of the Participating Institutions. Work is currently underway (sections V-C to V-E) to resolve the inconsistencies in conceptualization and tabulation that exist between the sites.

4. User Services and Support - User services and support in the model are combined as a single dollar value for each service type offered by every site. Each site must determine these levels after taking into account both demand and supply considerations. Thus the single dollar level per service type represents such diverse factors as written documentation, user consulting groups, CAI, audio-visual aids, and telephone information. It is assumed that the appropriate combination of these facilities is provided at every site for each service type. The shortcoming of this approach is that it loses selectivity based on type of support and assumes that quality is directly proportional to dollars expended. It has the advantage of allowing efficient quantitative comparisons.

a. Demand - The level of user services and support will affect the amount of demand at each site as well as its allocation among the various suppliers. Each user category may have a different degree of sensitivity (ranging from low to high) to services and support. Demanders view support as a single dollar level. This single amount comprises two types of support. The first is fixed support which is independent of usage and includes such items as development of manuals, on-line documentation aids, and consulting staff. The second

type, variable support, is dependent upon usage. This includes printing and distribution of manuals, free listings, and the use of on-line documentation.

b. Supply - On the supply side, each site must determine the total amount that it will spend on user services and support, and how that amount will be allocated among the various service types. Individual sites will place different degrees of emphasis on user services and support depending on their general profile. The model currently permits sites to distribute budgeted support levels across the various service types based on either the "unit of demand" or "relative dollar level" method (or a combination of these). The unit of demand method calculates service specific support levels on the basis of usage (i.e., divide the number of jobs or connect hours for each service type by the total number of jobs and connect hours for all service types at that site). This philosophy assumes that a small user requires as much center-provided help as a large user. The relative dollar level method computes service-specific support levels on the basis of dollar income from that service type (i.e., divide the gross income from each service type by the total gross income for all service types). If desired, sites can also specify particular dollar amounts to be assigned to selected service types. This would often be the case, for example, with new service offerings or services known to require disproportionately high (or low) levels of support.

D. Computer System Performance Modeling

1. Problem - The computer system performance modeling problem can be stated rather simply. There are two aspects:

a. Given a desired workload (job mix) to be processed at an installation, what is the performance of the system that the user perceives -- i.e., turnaround for batch jobs and response time for interactive applications?

b. What is the total capacity of the system to do "work?" That is, how many jobs of a given mix could it handle?

Both of the above questions are quite common and, in fact, must be answered to some degree for every new computer system or system modification. There are a wide variety of simulation, analytical, and empirical techniques available for these tasks[21]. Unfortunately, in this application the long time periods, the changing workloads, the number of different sites, and the poor definition of workloads rendered the standard approaches either inapplicable or computationally prohibitive. Fortunately, great accuracy is not required, and it was hoped that analytical techniques capable of providing adequate estimates could be developed.

2. Computer Unit Approach - Initital Attempts - Early analytical efforts focused on the definition of a so-called computer unit vector (c.u.). The c.u. vector's component values were to be the resource capacities available on each institution's computer system, such as the total CPU instruction executions available per unit time, memory residency requirements, paging, total card reader input capacity in cards per hour, and similar capacities for all other I/O processing and storage devices. It was initially hoped that a common computer unit vector could be defined which, for a given installation, would have component values equal to the capacity limit of each resource type at that site. The performance model would then require only one vector for each installation.

Central to the c.u. approach was the assumption that it would be possible to describe all network jobs in terms of a

small number of standard service categories. The model for each service category would include the resource requirements of a single "average" job. A given demand for computer services could consequently be expressed by the number of jobs in each service category. Resource requirements for the total demand would then be the sum over all service categories of the number of jobs in each category times the resource requirements per job. Thus, once the demand was established for a site, total capacity required could be determined and the corresponding supply in terms of that demand would be directly available from the c.u. vector.

Given the above descriptions, it would then be possible to develop formulas to estimate the response time for interactive services and the turnaround for batch services as a function of the utilization of computer system resources. The simplest formulas could be based on a model assuming a simple queueing facility at each site. Other possibilities considered for the proposed model included the use of more comprehensive queueing theories or other statistical models and, alternatively, algebraic formulas based on linear or least squares curve fitting to approximate graphical representations of response times.

A great deal of developmental and analytical effort was expended developing the above concept. With the assistance of the Stanford Research Institute[22], a computationally feasible implementation of this technique was developed. Unfortunately, incorporation of the results into the model would have required a prohibitive amount of data from the participating institutions, along with an extensive benchmarking study.

3. Revised Model - Current Implementations - Although the efforts towards developing a computer unit vector description of work were not successful in themselves, they did provide the framework for the simpler techniques that were finally adopted. The concept of standard service types (Section IV.C.3) is used exactly as developed in the above study and forms the cornerstone for representing levels of supply, demand, and

network flows. The vector of critical resources that is now carried serves a function similar to the c.u. vector described above. The difference is that, instead of an analytical derivation of turnaround, simple table look-up techniques have been substituted. The independent variable in each search is the most constraining of the vector components. The implicit simplifying assumption, therefore, is that the critical resources are independent and that secondary effects due to the interaction of resources can be neglected.

a. Turnaround - Batch turnaround is currently estimated as the sum of delays due to input, processing, output, and communications. For interactive response, only processing and communications are considered. Each of the participating sites provided empirical data in tabular form concerning batch turnaround and interactive response as a function of the weekly load on those system resources they considered critical. Since estimates of input and communication delays were consistently much smaller than those for processing and output (printer), the current model implementation only considers the latter two factors.

The printer (output) delay table describes a single curve representing the weekly average printer delay as a function of the degree of utilization of the printer. A simple FIFO queue discipline is assumed, and no differentiation is currently made between service types. Simple interpolation is used between the six stored table entries.

The algorithm for machine turnaround or interactive response allows for different levels of priority. By using different tables (one for each priority) and assigning one of six priorities to each service type, the model represents the difference in turnaround between the various service categories. Six sets of tables representing interactive response, fast or student batch work, and four levels of batch priority are included for each site. The number of terminals connected is the only critical resource

used for interactive work (priority one). For the five batch priority levels, up to three different critical resources can be included (e.g., CPU, memory, I/O channels, etc.). The current algorithm selects the most constraining of the three resources and uses it to estimate turnaround.

b. Resource Capacities - Each supplying site in the network simulation has an associated list of its constraining system resources. These resources include such items as cards read, print lines, communications capability (kilopackets), connect hours, CPU capacity, I/O operations (EXCP's in IBM jargon), and memory (kilobyte-hours). The capacities of these system resources are defined within the model as weekly figures. Weekly capacities are a function of scheduled "up hours," availability of the resource, and rated hourly capacity of the resource. Example: ABC university specifies that they can print at most 10,000 lines per hour; they have 100 scheduled hours per week; and their average availability is 90%. The theoretical weekly capacity for print lines is then: 10,000 X 100 X 90% = 900,000 lines/week.

While most resources can be defined in terms of weekly capacity, some resources must be viewed differently. Typical examples are main memory, on-line storage, and number of tape drives -- all of which are static constraints that do not vary with respect to time. Memory is partially handled on a per-job basis (i.e., will the job fit?), and partially by defining a time related unit such as kilobyte hours. The implications of limits such as the number of tape drives are much more subtle and are not accounted for in the present simplified model.

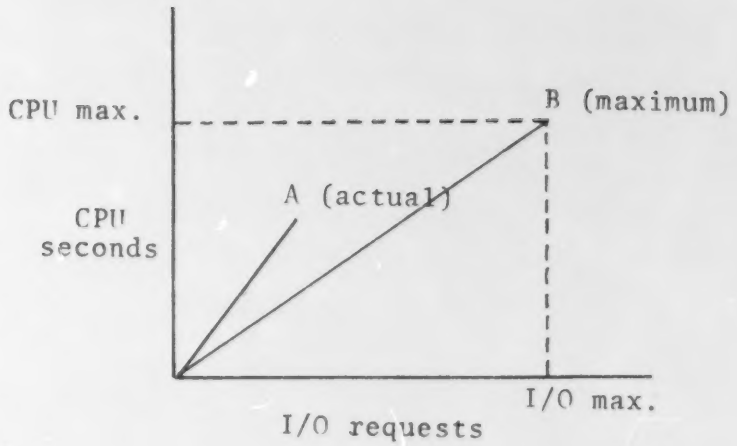
c. System Utilization - Utilization of each system resource is calculated within the model as the requested capacity per week divided by the total capacity per week. Requested capacity at a site is obtained by mapping all service specific demand to that site into resource requests. For example, a short batch statistical package job may map into 100 cards read, 2 CPU seconds, 250 print lines, etc. These resource requests are then summed over all requested work.

Total system utilization (per cent utilization of total capacity), is a crude, but useful, indicator of the appropriateness of the current job mix. A mix that is suited for a particular machine should lead to efficient system utilization and thus a low percent utilization of capacity. On the other hand, an inappropriate job mix (I/O oriented job on a CPU oriented system of the same size) will result in inefficient use of the system and a relatively high per cent utilization of capacity. Total system utilization is estimated as follows:

$$S = \frac{\sqrt{\sum_{i=1}^{m} (u_i^2)}}{\sqrt{\sum_{i=1}^{m} (c_i^2)}}$$

S = Total System Utilization
u_i = Actual usage of resource i
c_i = Capacity of resource i
m = Number of system resources

As a simplified example, consider a system with only 2 constraining resources: CPU seconds and I/O requests. Graphically, a plot of actual usage versus maximum (total) capacity might appear as follows:



Point A represents the actual utilization of both CPU and I/O, and point B is the maximum utilization of these resources. The ratio of the vectors OA and OB yields the percent utilization of capacity.

d. System Saturation - If any constraining resource becomes very highly utilized, the total system will correspondingly tend towards saturation and performance will be affected. This is true even if, due to an inappropriate job mix, system utilization is still relatively low. Thus, system saturation is defined as the percent utilization of the most highly utilized resource on the system.

Note that percent saturation will always equal or exceed percent utilization. The difference is an indication of the inefficiency of the job mix -- i.e., they are equal only for an "optimal" workload. Pricing algorithms and other policy decisions should therefore tend to reconcile any differences between the two vectors. Turnaround estimates must be based based on percent saturation (this is mathematically equivalent to the procedures described in paragraph b above).

4. Network Queueing Models - Future Model Enhancements - Recently, there has been considerable interest in the application of network queueing theory to the performance modeling of computer systems[23-33]. New computational algorithms have been developed which drastically lower the amount of computation necessary[27], and it has been shown that accurate analytical models can be developed for most computer systems[34]. One author even foresees generally accepted axioms of computer performance prediction[29]. Network queueing model techniques have been applied to two of the sites in the study (MIT and Dartmouth), and the results thus far seem encouraging. This type of model has an advantage over the present empirical implementation in that the effects of hardware changes on throughput and response times can be easily modeled. It is expected that later phases of the project will use these methods of performance prediction in some capacity. It has not yet been determined whether they are efficient enough to replace the current table look-up procedures completely or should be used only for periodic updates to these tables.

a. Overview of Network Queueing Theory - The approach used in network queueing theory is to consider the computer facility as consisting of a multiple server queueing system, where "tokens" (programs, jobs, transactions, etc.) pass from one server to the next, and then back to the first server. This cycle is repeated a number of times for each transaction,

with potentially more than one transaction cycling between servers simultaneously. In the theory's simplest implementation, there are only two servers, CPU and I/O. Each transaction consists of a number of iterations between the CPU server and the I/O server. For instance, a file update job might consist of 300 input-output operations with each operation taking 30 milliseconds and the interval between operations (the CPU service time) approximately 300 milliseconds. This job would take (300*0.030) + (300*0.30) = 99 seconds to complete on an empty system. CPU utilization would be (90/99*100) or approximately 90%. However, what if two of these jobs were run simultaneously? Once multiprogramming is allowed, queues can develop at both the CPU and I/O servers. The situation becomes even more complicated when we have differing types of transactions (i.e., with different I/O and CPU service times), or when there are several types of I/O devices available, each with different characteristics. Problems of this type can be solved analytically using network queueing theory.

b. Current Status of Project Cyclic Queueing Research

The internal modeling subroutines necessary to evaluate both the MIT and the Dartmouth systems have been completed and are operational. It is hoped that these routines can be used for several other sites with little modification. The parameters necessary to drive the MIT model have been obtained and validated. Predictions of CPU utilization are within 1%, predictions of interactive response time within 3/4 second, and predictions of turnaround time are within 8 minutes[35].

c. Incorporation into the Simulation Model - Incorporation of the central server queueing algorithms into the Network Simulation model will have three effects on the current model:

First, the table-driven turnaround modules currently being used will be supplemented or replaced by site-specific subroutines which evaluate the queueing model equations for the individual site. Thus there will be additional code required to perform the calculations, as well as some increased run time to execute the code.

Second, there are some additional data required from each site. Data are needed describing transition probabilities as jobs move from the CPU server to a particular input-output server (i.e., disk, tape, drum). Also the device service time is needed for each of the devices on the system. An estimate of the number of jobs concurrently resident in main memory is also necessary. All other information needed by a cyclic queueing model has already been obtained.

Third, additional code has to be written to input and to store all the additional information required and produced by the performance subsystem. Current estimates are that this could be as much as 15,000 characters of storage for each site if each site needs a separate performance module. This can be reduced dramatically if sites can use the same code.

In summary, the use of a cyclic queueing model could significantly improve the performance estimation capability of the model. Unfortunately, the price of this improvement is an increase in run time and

storage requirements. Tests have not yet been made with the full simulation model. If the increases prove to be minimal, it is expected that the queueing models will completely replace the current table-driven turnaround modules. At the other extreme, if the run time and storage penalties prove exhorbitant, the queueing models would not be integrated into the main system. Rather, they would be used periodically off-line to compute the parameters of the tabular functions. A likely compromise is that they would be integrated into the model, but only used when the tabular functions required updating..

E. Site Representations

Since the model is intended to reflect the characteristics (actual or trial) of different institutions, it must be capable of representing these with site-specific data. The site representations are kept as input files so that changes to a site's data need not involve changes to the model itself. This technique also allows simulation runs to be made with different numbers and combinations of sites. The areas unique to each site are summarized below and are discussed in more detail elsewhere.

1. User Categories - The demand at each site is viewed as consisting of from one to ten user categories, each with its own behavioral characteristics. These are specific to each site though many sites have selected similar groupings. Typical examples include: administrative users, student instruction, and research. These are groups for which separate budgets are provided and different usage policies may exist. Aggregate demand is developed at the user category level and is then mapped into its service type components(i.e., the number of jobs or connect hours of each service type). This demand is then allocated to sites depending upon the policies currently in effect.

Each user category at a site may have a different sensitivity to price, turnaround, support, and other factors, reflecting the varying importance which each user may give these items. Each user category may also have different seasonality factors which reflect changes in demand levels over the course of a year. (For example, student use generally decreases during the summer.)

2. Hardware/Software Representation - Each site has associated with it a vector containing the capacities of its "critical resources." This vector represents the maximum usable capacity of each resource over a week. (Usable capacity is defined as the average available capacity, not the theoretical maximum). Each service type offered at a site has comparable vector of coefficients associated with it which contains the amount of critical resources required for each job or connect hour of that service type. (One unit of a service type is equivalent to one job for batch service types and one connect hour for interactive service types).

In addition to indicating which service types are available and the support it provides for each of them, the site must also specify a communications limit for network transmission, the number of scheduled hours it is available for use during a week, and its average availability (i.e., % reliability).

3. Policies - Each site's decision making is modeled by choosing different policies which it is to follow. These fall into three areas: demand, supply, and market.

   a. Demand Policies - These determine the workload desired at a site for a given period and where this workload should be processed. The demand policies are used to choose the relative site sensitivities of demand allocation to price, turnaround, support, momentum, and user budget restrictions.

b. Supply Policies - These cover two areas, supply determination and supply allocation. Supply determination decides the amount of hardware, software services available, budget dollars, prices, and support levels. Each of these has several policies available which may be selected by an individual site. Supply allocation policies describe the allocation of supply at a site to the various users demanding the available facilities. For example, one site may give instructional users first choice of available services. These allocation policies become extremely important at heavily loaded sites.

c. Market Policies - These policies determine the action to be taken if all the requested demand cannot be satisfied. They indicate the method by which cutbacks are accomplished.

4. Special Representations

a. Discounts - Some of the most difficult modeling problems encountered involve the special agreements among sites. For example, one site may have a discount arrangement with another site. This allows the first site to buy computer power from the second at wholesale prices in return for volume or other guarantees. In the model, discounts are allowed between any two sites by means of a discount matrix whose elements are a multiplier for the normal price at a site.

b. Communications - As volume between two sites increases, it may become cheaper to lease a communications line between the sites rather than use a network. The communications matrix contains a factor which is multiplied by the volume between

any two sites. It is relatively easy, therefore, to add policies stipulating when a site will use the network and when it may choose to lease a private line.

F. Supply Determination and Estimation

One of the primary concepts in the model is that of the supply of computer services. Supply is more than just a source of raw computing power, CPU cycles, and available disk drives. It also represents the services which are available to users of a system, and the support which these users may receive to facilitate their use of these services. The supply modules determine the amount, type, cost, and quality of computing services offered at each site.

The supply segment of the model (module 3.2) is policy-driven. After initial specification of its hardware and software offerings, budget, prices, and user support, each site may select policies and practices which represent its actual (or test) decisions in these areas.

1. Budget - A factor which underlies all elements of supply is the budget. Once the budget has been determined at the administrative level, specific decisions on the lower levels of hardware, services, user support, and prices may be made. Various budget policies can allow changes to budget items, reallocation of funds, and various other overall adjustments. Specific policies may then determine what is done with the budget funds. For example, if a hardware increase is indicated, budget policy may require that sufficient funds be immediately available to cover the expense of upgrading, or it may only specify that the projected increase in revenues be sufficient to cover the increase in monthly rental fees. The supply of services can also be controlled by the budget. If a new software service is to be added, it may be necessary to have funds available to cover the implementation costs. User support is based on budgeted funds, and its allocation

depends on the policy chosen.

2. Hardware - Rather than represent each specific item of hardware, it was decided to concentrate on the resources available at a site. That is, the actual hardware and system software items (vendor name, model number, release number) are described in terms of more basic units of computing power (print speeds, memory size, CPU capacity, levels of multi-programming) that are relatively site-independent. Hence, addition of a model 1234 printer of brand HAL might be described in model terms as an incremental 1,200 lines per minute of effective print capacity.

3. Software Services Available - In addition to computing power, sites also offer a variety of services (Section IV.C.3). The computing capacity determines constraints on what and how much a site can and cannot do. However, the user (demander) is only interested in the software services (service types) that are available to him. Network flow is represented in terms of these services and a site's attractiveness is based on which services it offers and the conditions under which they are offered. This supply element can be increased or decreased depending upon a site's policies relative to the addition or dropping of particular service types.

4. Support - Another "supply" item which is offered by sites is user support. This is a less tangible, but important, factor which affects computer usage. It includes written documentation, user consulting groups, computer assisted instruction, audio-visual aids, and other assistance which an installation may provide to users of its facilities. Although this is not always a specific item on a budget, it is a real expense. Some sites may choose to put little money into user support and instead to concentrate on improving their software offerings. Others may decide to provide support money to services in proportion to the income which these services generate. Each site must make decisions such as these relative to the level and distribution of

"supply" of user support over the software services which it offers.

5. Prices - Although price is not a supplied item per se, it is a major factor in determining the allocation of this supply. The supplier of services must specify the method for charging for these services. Usually, prices will be based on the raw resources utilized by each job, although some sites may want to set prices at the service type level. More difficult than the setting of prices is the specification of policies for that setting. I.e., what level of cost recovery is desired, should selected services be reduced in price in order to encourage usage, should user services be included in the list of raw resources upon which price is based?

G. Demand Estimation

A second fundamental concept in the model is that of demand. In order for a site's hardware and software to be utilized, there must be a demand for these offerings. The demand module (3.3) has three basic segments. First is the overall policy interpretation which controls the remaining processes. The demand generation module then estimates the actual totals of computer services desired, and the allocation module determines where the users go to satisfy their needs. The demand generated by the model should approximate that of the sites, reflecting both the types of users and the types of jobs that they require. In order to represent these two areas, the concepts of site-specific user categories and network standard service types have been employed in the model (Section IV.C). Both demand generation and demand allocation is first handled on an aggregate user category level, then broken down by specific service types.

1. Demand Estimation - Current demand is estimated for each user category at a site based on such factors as previous demand, expected growth, and current turnaround, price, support, and seasonality. The growth factor is specific for each user category and reflects the estimated overall growth profile. Each

time period, the expected growth is first added to the previous base level of demand. This base figure is then modified based on the user category's sensitivity to turnaround, price, and support, and the current levels of each of these factors. The demand is then scaled by a multiplicative seasonality factor which takes into account the monthly variations that might occur in many user categories. Finally, the cost of satisfying this demand is estimated, and any cuts necessitated by budget limitations are made.

Once the demand of each user category has been established, this value is expanded into demands for individual service types. The service types demands are then aggregated over all user categories at a site so that total demand can be expressed directly in terms of potential network flows. This potential total demand must now be allocated to available supplier sites (including in-house).

2. Demand Allocation - Selection of supplier sites for each user category first requires a determination of eligible sites. This is based on local user category restrictions (policies), as well as expressed willingness of supplier sites to serve that institution. Then, for each service type, a rating is developed on a scale of zero to one for each candidate supplier. The coefficients used in these rating equations give various degrees of emphasis to factors such as price, turnaround, support, and past usage of that site. Note that this rating must take place at the user category level in order to reflect differences in policy, constraints, and behavior patterns among the various users.

Two additional factors allow for in-house preferences and "stickiness" or switchability. The model parameter DEBUMP, for in-house preferences, is a scaling factor on the site's internal rating to make it appear more (or less) attractive than the other sites on the network. The momentum factor affects switchability. It is set to the full value for sites which satisfied demand for

this service type last period, and is set to zero for all others.

Currently, the rating equation is a linear sum of all factors listed above, each multiplied by the appropriate coefficient. More sophisticated algorithms could easily be accommodated. All candidate sites are ranked based on this rating, and demand is allocated to the top "n" sites in the ratings (n is the demand allocation variable in the overall policy vector and indicates the number of different external sites over which demand for a given service type can be allocated).

The "n" sites with the highest ratings are allocated the demand for that service type in direct proportion to their relative ratings. (Only sites which have a higher rating than the demanding site are included. Thus, if the demanding site has the highest rating, it will be allocated all of its own demand for that service type.) Some demand will always be allocated in-house if the service type is offered there.

H. Market

Although a supply of computer services may be available and a demand for these services may be generated and allocated, there is no guarantee that all of the demand can be satisfied by the selected supplier sites. The market module determines how much of the demand can be satisfied and how to cut back if required. At a given site, the rules by which demand is satisfied or not satisfied are actually combinations of policies and system scheduling strategies. In the simulation model, this complicated procedure is reduced to a set of policies which may be applied when cutbacks are necessary. The policies are explained in detail in Appendix II-F.

I. Financial Considerations

1. Budget and Cash Flow - Early in the model design, it became clear that a comprehensive representation of each site's

budget would have to be maintained. It would be difficult if not impossible to model decisions which affect financial flows without knowledge of the amount of money available. This is further complicated by the multiple representation levels in the model. Budget information is thus carried at three levels in the model.

First, user categories have budgeted amounts for expenditures on computer services. Second, the computer center has budget items determining its hardware, communications, and personnel expenses. Finally, there are budget estimates of income from local users as well as network users. Affecting each of these areas may be overall restrictions (policies) imposed by the central administration on such items as minimum permissible net balance of trade (below which "outside" purchasers must be restricted), cash flow, or "profit."

In order to collect the budget data, every site participating in the simulation study submitted annual budgets describing the major income and expense items as defined and requested in Questionnaire II (Appendix VIII). These are used to produce the site budget report (Figure IV-2). Currently, budgets are based on a yearly time interval starting with the first week of the simulation. Policies used with the experiments performed thus far assume that net bottom line budgets (i.e., total income and total expenses) do not change within this yearly period. However, reallocations of available dollar amounts among existing budget lines are permitted. Planned future policies will permit revisions of forecasted income and expense items based on results and trends to date. In order to track income and expenditures, the model can provide each site with a cumulative status report each week (Figure IV-3).

The cash flows can easily be projected to cover a yearly period, so that discrepancies between budgeted and actual expenditures can be examined. The specific manner of comparison (i.e., this week vs. 1/52 annual; f(total to date, weeks remaining); etc.)

Figure IV-2

```
SITE BUDGET REPORT                                  REQUESTED BY: ABC

NETWORK SIMULATION                                  RUN DATE:  2/19/76

SITE  ABC                                           WEEK:   0

ANNUAL BUDGET                                       AS OF WEEK   0


EXPECTED INCOME:
----------------
   INTERNAL USERS                          $      300000.   ( 69.77%)
   EXTERNAL (NETWORK) USERS                        10000.   (  2.33%)
   OTHER (GRANTS, ETC.)                           120000.   ( 27.91%)
                                           ---------------------------
   GROSS INCOME                            $      430000.   (100.00%)

COMPUTER CENTER EXPENDITURES:
-----------------------------
   HARDWARE/SOFTWARE                       $     1138468.   ( 20.99%)
   FUNDS FOR IMPROVEMENT                           45000.   (  0.83%)
   COMMUNICATIONS:
     FIXED                    15000.
     VARIABLE                 30000.       $       45000.   (  0.83%)
                              ----------
   OPERATIONS STAFF                               263451.   (  4.86%)
   PROGRAMMING STAFF                              635384.   ( 11.72%)
   USER SUPPORT                                   124627.   (  2.30%)
   SUPPLIES                                       394792.   (  7.28%)
   ADMINISTRATION                                 302882.   (  5.58%)
                                           ---------------------------
   TOTAL COMPUTER CENTER EXPENDITURES      $     5423484.   (100.00%)
                                           ---------------------------
   NET COMPUTER CENTER LOSS                $(    4993484.)
                                           ------------
                                           ------------


USER COMMUNITY EXPENDITURES:
----------------------------
     USER CATEGORY   I          107533.
     USER CATEGORY   II         620439.
     USER CATEGORY   III         61989.
     USER CATEGORY   IV        1228120.
     USER CATEGORY   V           54138.
     USER CATEGORY   VI         401661.
                              ----------
   TOTAL USER EXPENDITURES                 $     2473880.
   LESS: INTERNALLY SPENT FUNDS            (      300000.)
                                           ------------
   NET (EXTERNAL) USER EXPENDITURES        $     2173880.
                                           ------------
                                           ------------
   U OF GEORGIA NET BALANCE PERIOD  0  $(    7167364.)
                                           ------------
                                           ------------

   *** BALANCE OF TRADE =          0.
```

Figure IV-3

INCOME/EXPENSE REPORT (CUMULATIVE) REQUESTED BY: XYZ

NETWORK SIMULATION RUN DATE: 7/23/76

SITE= XYZ WEEK: 20

TOTAL INCOME TO DATE:			
INTERNAL USERS		$ 131099.	(81.67%)
EXTERNAL (NETWORK) USERS		6338.	(3.95%)
OTHER (GRANTS, ETC.)		23077.	(14.38%)
GROSS INCOME		$ 160514.	(100.00%)
COMPUTER CENTER EXPENDITURES TO DATE:			
HARDWARE/SOFTWARE		$ 178415.	(47.71%)
FUNDS FOR IMPROVEMENT		0.	(0.0 %)
COMMUNICATIONS:			
FIXED	$ 6308.		
VARIABLE	370.	$ 6678.	(1.79%)
OPERATIONS STAFF		85495.	(22.86%)
PROGRAMMING STAFF		18891.	(5.05%)
USER SUPPORT		25248.	(6.75%)
SUPPLIES		40125.	(10.73%)
ADMINISTRATION		19134.	(5.12%)
TOTAL COMPUTER CENTER EXPENDITURES		$ 373985.	(100.00%)
NET COMPUTER CENTER LOSS		$ 213471.	
USER COMMUNITY EXPENDITURES:			
USER CATEGORY I	137995.		
USER CATEGORY II	58871.		
USER CATEGORY III	76361.		
USER CATEGORY IV	52335.		
USER CATEGORY V	65324.		
USER CATEGORY VI	10733.		
USER CATEGORY VII	33340.		
TOTAL USER EXPENDITURES		$ 412324.	
LESS: INTERNALLY SPENT FUNDS		(131099.)	
NET EXTERNAL EXPENDITURES		$ 281224.	
TEXAS TECH NET BALANCE PERIOD 20		$(494696.)	
*** BALANCE OF TRADE		$(274887.)	

86

is a site-dependent function of policy.

Budgets may affect levels of both demand and supply at a site. In cases where a site is following a policy of free spending, for example, budget limits may have little effect. At the other extreme there are many policies which force strict compliance with all budget constraints and user categories might be restricted in their demands due to monetary limits with respect to expenditures. In cases of high site utilizations, sites may upgrade their hardware/software configuration if sufficient funds are available, but might look to other alternatives (impose restrictions on network inflows of demand, etc.) if funds are lacking.

2. Pricing and Cost Recovery - Pricing is a critical topic in the study as many of the experiments proposed involve varying pricing practices, policies, and regulations. The participating institutions have a wide variety of pricing algorithms and cost recovery mechanisms, and it was difficult to develop an accurate representation of each. The approach followed is parametric in nature and permits a variety of algorithms and policies to be represented using the basic structure described below.

Internally, the model carries a single unit price associated with each service type. This may be derived from resource usage or set explicitly. The sites provided, in Questionnaire II, a list of "critical" resources. For each service type, the resource consumption of one job or terminal session is fixed throughout the simulation. Therefore, knowing the charge (per unit) of each resource allows the computation of the price per job as the linear sum of the components. Consequently, site pricing policies may focus on the individual resources (letting the model determine service type prices), or they may function directly at the service type level.

As a result of data compression and manipulation, the prices computed for various service types tended to be slightly different

from the actual prices reported in Questionnaire II. Additional discrepancies can be attributed to the use of only eight critical resources in the calculation of job requirements and costs. (Some sites had over 50 components in their actual billing algorithms). While excluding non-critical resources made little or no difference in the performance modeling, prices for some services varied considerably from the stated price. The approach taken was to create a pseudo-resource labeled "all other," the price of which was set equal to the difference between the actual and model prices. This allowed the tuning of prices until they matched the reputed levels.

Another factor which had to be considered was priority pricing which is used at many sites. In this situation, a job with identical resource requirements could have several different prices depending upon the priority which it had been given. In order to adjust the prices for different levels of priority, a second pseudo-resource, which indicated the priority level and acted as a multiplicative adjustment factor, was added to the computation.

J. Communications

An essential ingredient for computer network resource sharing is the existence of data communications facilities which can reliably transport data between network sites. The domestic switched telephone network, though designed for voice communications, can also be used for data communications. However, when used for data, the voice network possesses limitations including reliability, noise, and capacity. The voice facilities are often used very inefficiently when carrying data and consequently its costs are high.

In the late 1960's, a project of the Advanced Research Agency of the Department of Defense was undertaken to explore and implement a new communications technology designed specifically for data rather than voice. This project, known as ARPAnet, used a

technique called packet switching to demonstrate the technical and economic feasibility of reliable, error-free, high capacity data communications service. There are currently many computer systems at universities, research organizations, and governmental agencies connected to the ARPAnet. Technically, the ARPAnet is quite similar to the inter-institutional resource sharing network being modeled in this project.

The ARPAnet communications technology has been transferred to the commercial sector by Telenet Communications Corporation, a private firm. A functionally similar service is offered on TYMNET by TYMSHARE, Inc. These communications offerings, called Value Added Networks (VAN), have the following characteristics:

1) widely distributed geographically
2) high reliability due to redundant communications paths
3) code conversion and terminal handling procedures for a wide variety of low speed interactive terminals
4) error detection and recovery procedures
5) a pricing structure based on usage and not a function of distance

Since these VAN services are now available in the competitive marketplace, the prices charged can be considered as costs in the model.

The communications cost structure and parameters in the model are based on the Telenet tariff. The specific cost structure in the model consists of the following components:

1) one-time setup or conversion cost
2) fixed monthly cost
3) variable cost directly related to volume of data

The one-time cost represents any modifications to the host oeprating system or communications front required to interface to the communications network. For the hosts at participating

institutions this cost estimate ranges from zero to $25,000. The zero cost applies to hardware for which the communications vendor makes the necessary software available free. In general, it is easier to interface a communications front-end processor, for example, than a host mainframe. These differences are reflected in the cost estimates. It is possible in all cases to avoid this one-time cost entirely if the host system will only be using the network for a restricted set of service types. For example, the one-time cost can be avoided (but the monthly fixed cost is increased slightly) if that site restricts its supply and consumption of network services to include only interactive service types. Support for remote batch service types requires that the one-time cost be incurred.

The fixed monthly cost component represents the costs incurred for leased channel from the supply or consumption location to the nearest network entry point. There is also a monthly charge at the entry point for the network port dedicated to the site. This fixed cost component varies depending on distance between site and entry point and the capacity of the channel. For a site located within a few miles of the nearest entry point, using a channel with a 2400 bit per second capacity, the monthly fixed cost would be approximately $750. The capacity could be quadrupled for an additional $400/month. The most costly network interface of all sites in the project would be $1600 and $2000 per month for a channel capacity of 2400 and 9600 bits per second, respectively.

The variable cost component represents the cost which is entirely dependent on the number of characters (or packets) which are carried by the network. The cost parameters currently used for experiments is 60¢ per thousand lines or cards with batch service types, and $3 per connect hour for typical low speed interactive service types. The commercial communications network has virtually unlimited capacity. However, the channel between the site and the network entry point has a clearly defined

capacity. Each service type in the model has a defined requirement (per job or per connect hour) for this access channel resource.

While the cost component parameters described above are in terms of Telenet technology and tariffs, the cost structure in the model can accommodate many other current communications services. For example, the telephone companies and specialized carriers like DATRAN offer communication services which may, under some circumstances, be more attractive for some high volume sharing relationships. Representing the tariffs and capacities of these is easily accommodated by the present model structure.

Present values of communications capacity were assigned based on estimates of potential work flow by the project staff. These values will be refined as experience is gained with the model. Note also, that communications capacity is one of the resources that can be changed automatically as the need arises if a policy defining permitted changes is provided.

V. DATA COLLECTION AND ANALYSIS

A. Overview

During Phase I the principal method for collecting data from participating institutions was the use of written questionnaires. Since the tasks of data collection and model development were conducted in parallel, it was first necessary to develop a relatively unstructured questionnaire (Section V.B) which would permit respondents at participating institutions to report on general site characteristics such as available hardware and software, the nature of the user population and its demands for services, and the financial and organizational characteristics of each site. At this early stage in the model development, responses to Questionnaire I were useful in defining model structures which could represent the diversity of the participating instructions.

As model development proceeded, detailed structures for representing sites and inter-site traffic were developed to reflect the response patterns in the first questionnaire and the needs of the simulation model. Since the initial responses from the sites did not, in general, match this model structure or satisfy all of the data needs, a second questionnaire (Section V.C) was developed to obtain quantitative data in a form compatible with model needs. The major area where such compatibility is essential is the definition of service types (Section IV-C.3) which can flow in the network. Most of Questionnaire II was devoted to the estimation of demand and capacity in terms of the uniform service types.

The responses to these two questionnaires provide the basis for the modeling of supply and demand at each site and also set the initial conditions for the beginning of most network simulation runs (i.e., what the site looks like without networking).

Since services, rather than raw machine cycles, will flow in the network, each service type must have an associated unit of measure - - i.e., a standard job. One possible approach would have been to develop a benchmark program for each of the over forty different service types. However, the running and tabulation of these benchmarks at each network site would clearly have required an inordinate expenditure of time and resources. Further, the problem of describing individual site workloads in terms of the standard jobs would have been virtually impossible. The approach taken (Section V-D) took advantage of a coordinated activity with the Planning Council in which a series of FORTRAN programs were executed at a number of sites, including many of the participating institutions. (A subset of this series of tests was run at those participating institutions that are not members of the Planning Council.) These benchmark results are being used by the project team to reconcile differences between the job types reported by the sites and the service types used by the models. They also provide a formal set of comparisons between sites relative to pricing and resource requirements of identical jobs. Of perhaps greatest interest, they give a quantitative indication of the wide variation in prices and service between institutions. Even though it is clear that much of this variation would be diminished in a network environment, the potential benefits and resource sharing that this study uncovered are very great.

B. Questionnaire #1

The first Phase I questionnaire was distributed to representatives of the participating institutions at a series of three one-day regional project orientation meetings held in October, 1975. About one-third of each meeting was devoted to discussion of the philosophy behind each major section, as well as the specific information requested. It was considered essential that site participants understand how their responses were to be used in the simulation model, so that the widely varying hardware, software, and organizational structures could be adequately represented. After the regional meetings, many telephone conversations and letters were used to

clarify how questionnaire responses should reflect unique site characteristics. A particularly troublesome aspect of the questionnaire was the definition of categories to represent computing services supplied and consumed at the various sites. It was initially felt that no single set of categories would be meaningful to all sites and still satisfy the model requirements. Therefore, the resulting questionnaire presented an illustrative categorization to indicate the level of detail desired, but respondents were expected to create categories that were appropriate at their respective sites. These site-specific categories formed the basis for deriving the network standard categories used in Questionnaire II.

An outline of Questionnaire I appears as Figure V-1, and the full questionnaire is included as Appendix VI in Volume II of this report. The eight sections of the questionnaire are discussed below:

1. Nature and Supply of Computing Services - This section was intended to obtain a description of the hardware and software resources available at each site, and any constraints and conditions relative to using and charging for these services. It requested a description of those facilities which might be of interest and available to outside network users. Questions were also included about current practices and policies relative to adjustments in capacity and offerings.

2. Demand for Computing Services - This section investigated the demand for computing services both at present and in a potential networking environment. In particular, it requested the categorization of users in a way that would be meaningful for budget and policy making decisions. The institution was then asked to provide a general profile of the workload for each user category and also information about usage by job type. Finally, an attempt was made to obtain an initial description of policies on outside usage that might affect the way the institution would deal with a national network.

Figure V- 1

Questionnaire I Outline

I. NATURE AND SUPPLY OF COMPUTING SERVICES

- A. Hardware and Systems Software
- B. Software Products/Resources
  - 1. Service Offerings
  - 2. Constraints on Service Offerings
- C. Prices and Charging Structure
- D. Candidates for use by Network Users
- E. Supply Practices and Adjustments
  - 1. Major Modifications in Capacity or Offerings
  - 2. Present Utilization and Surplus Capacity
  - 3. Minor Capacity Modifications
  - 4. External Users
  - 5. Capacity Reductions
  - 6. Internal Dedicated Systems

II. DEMAND FOR COMPUTING SERVICES

- A. User Categories
- B. User Characterization
- C. Present and Projected Demand by User Category
- D. Present and Projected Demand by Service Type
- E. Present Demand by Internal Users for External Resources
- F. Potential (latent) Demand from Internal Users
- G. Policy on Outside Usage

III. USER SERVICES

- A. Internal Users - Internal Facilities
- B. External Users - Internal Facilities
- C. Internal Users - External Facilities

IV. ORGANIZATION OF COMPUTING ACTIVITIES

V. INSTITUTION CHARACTERISTICS

VI. BUDGET

VII. RESOURCE SHARING ARRANGEMENTS AND POLICIES

VIII. OTHER

3. User Services - This included those products and/or services that enable a user to learn of the existence, characteristics, suitability, and usage procedures for computing services, as well as obtaining help as needed. Several general questions were included to gather information about this area.

4. Organization of Computing Activities - This section was designed to provide a feel for the decision structure and responsibilities at each institution.

5. Institution Characteristics - Information was requested about the overall nature of the institution - its disciplines, numbers of faculty and students, research orientation, etc. In most cases these data were provided from existing documentation.

6. Budget - Each institution was requested to provide a summary of its overall budget and that part of the budget allocated to computing activities. Questions were also asked about cost recovery policies so as to provide a guide to the relationship between prices charged to users and actual costs for providing the service.

7. Resource Sharing Arrangements and Policies - Each institution was asked about its present participation, if any, in existing consortia, cooperative arrangements, or networking.

8. Other - This open-ended section asked for information on constraints or commitments which might have an impact on an institution's computing activities but were not previously described. This included such things as state laws or purchasing regulations, budget limitations, and auxiliary enterprises such as hospitals.

C. Questionnaire II

In April, 1976, a joint effort of the Stanford Research Insti-

tute and the EDUCOM staff resulted in the second questionnaire. It was developed after completion of the model design and unlike the general nature of Questionnaire I, it sought to satisfy specific requirements of the model and the background studies. Consequently, recipients were provided with very specific response formats (Tables to be completed) to facilitate both their response to the questions and the processing of data after the questionnaires were returned. Although some of the material was covered in the first questionnaire, the responses to that document generally lacked the detail that was needed to meet the requirements of the model. Also, in responding to the first questionnaire, not all sites presented data in consistent formats.

Questionnaire II was divided into eight sections, each containing a table or tables for entry of specific quantitative data. The full questionnaire is included as Appendix VII in Volume II. The eight sections of the questionnaire were:

1. Resource Types and Capabilities
2. Scheduling Periods and Durations
3. Performance Characteristics
4. Response Time for Interactive Use
5. Weekly Resource Usage by Service Type
6. Job Distribution by Service Type and by Priority
7. Job Distribution by User Category
8. Budget

1. Resource Types and Capabilities - Each institution was asked to identify significant billable and/or potentially limiting resources and to estimate the usable (not rated) capacity per hour of each of these. Particular care was taken in defining the meaning of capacity. For example, the average hourly billable CPU capacity is typically much less than the maximum number of CPU cycles delivered in an hour. Up to eight resources, such as memory, CPU utilization, and output volume, could be selected by each site. After the responses were reviewed, it was decided to designate the

six most commonly selected resources as "standard." These are: Memory Capacity, Cards Read, Lines Printed, Connect Hours, CPU Seconds, and I/O Capacity. To this list was added a seventh standard resource, Communications Capacity. Although not yet a factor at most sites, this area is expected to be critical in a network environment.

2. Scheduling Periods and Durations - Site schedules were needed to define activity and resource availability during the week. The institutions were asked to list the maximum fraction of each resource available for interactive use (For example, only part of main memory might be available to interactive users). In addition, they were asked to provide their usual daily peak times for batch and interactive work (if different).

3. Performance Characteristics - In a network environment, all activity might not occur at the same site (i.e. input and output are local, while processing may be scattered). It is therefore necessary to provide separate estimates of input delays, processing tires, transmission times, and output delays (including queueing and other related delays). Total turnaround as seen by the user is the sum of all delays. Each site was asked to describe critical input resources and up to three critical processing resources. The limiting output resource was specified as the printer. For various percents of the effective practical capacity of each resource, the sites provided the associated delay time. Those delay times were requested by type of jobs, priority, and time of submission (peak or non-peak). Non-suppliers were asked to provide these data for input and output and to indicate the average processing time at their major source of supply.

4. Response Time - The stated assumption for this question was that interactive response time is primarily a function of the average number of terminals in use. Each site was therefore asked to provide tables describing this relationship.

5. Weekly Resource Utilization by Service Type - After the total workload (number of jobs of each service type offered) is estimated by the model, the resource requirements for each service type are used to calculate total system loading. In order to provide empirical data for these computations, the sites were requested to provide the total actual usage for each critical resource listed earlier and to break this total usage down by service type. This provided a profile of resource usage for the "average" job in each service type. If a site chooses a pricing policy based on resource usage, the model will be able to calculate the price per job from the per job resource requirements and the unit price for each critical resource. As mentioned in Section IV.H, an adjustment factor for resources which have not been included ("all other") must be added to arrive at the final job cost. An abbreviated sample of this table is provided below:

A. Resource Type	CPU	Input	Output	
B. Units	Secs	Cards	Print Lines	
C. Total Weekly Capacity				
D. Breakdown by Service Type: Restricted Resource Allocation				
1. WATFOR, WATFIV				
2. WATBOL				
3. PLC				
4. SPITBOL				
5. Fast Assemb. (ASSIST)				
Batch General				
6. Student FORTRAN				
7. Student COBOL				
8. Student PL1				
9. Student OTHER				
10. Prog. Dev. FORTRAN				
11. Prog. Dev. COBOL				

6. Job Distribution by Service Type - In order to estimate the cost of an average job of each service type with priorities taken into account, the sites were requested to provide the total number of jobs or connect hours per week of each service type. These totals were then broken into the fraction at each priority, the fraction at each priority submitted during peak time, and the cost of an average job at each priority. A sample of this table follows:

Service Type	Jobs Per Week	Priority 1			Priority 2			Priority 3			Priority 4		
		f_1	p_1	$\$_1$	f_2	p_2	$\$_2$	f_3	p_3	$\$_3$	f_4	p_4	$\$_4$
6. Student FORTRAN													
7. Student COFOL													
8. Student PL1													
9. Student Other													
10. Prog. dev. FORTRAN													

f_i = fraction of jobs with priority i

p_i = Fraction of priority i jobs submitted during peak time

$\$_i$ = cost of average priority i jobs

7. Job Distribution by User Category - Since many policies and budget constraints are based on a site's perception of its user categories, it was necessary to obtain the distribution of demand for the various service types over the institution-specific user categories. For example, externally funded research projects may be allowed access to the network but student instructional usage may not. After reviewing the first questionnaire, the project staff developed a tentative set of user categories for each institution. These were used in the second questionnaire and were generally accepted by most sites. The weekly job (or connect hour) count by service type was then distributed over the user categories.

(Note that the job counts by service type should match those of the previous table). An example of this table is provided below:

	Current Demand	User Cat.1	User Cat.2	- - - -	User Cat.10
6. Student FORTRAN					
7. Student COBOL					
8. Student PL/1					

8. Budget - Budget figures provide the data for many of the policy decisions and financial reports. The user category budgets, for example, indicate potential limits on spending.

Annual budget amounts were requested including internal and external income, expenses such as hardware and software, user support, supplies, operations and programming staff, and administrative expenses. Estimates of budgets or expenditures for each user category for that site were also obtained.

The data collected by both questionnaires were reviewed for completeness and consistency before being put into machine readable form for use by the model. The results of this review is described in Section V-E.

D. Benchmarks

Implicit in the concept of networking is the need for a qualitative basis for investigating cost and performance differentials across a variety of computing facilities. These two areas are important since they are among the major factors influencing the success of inter-institutional networking. The staff of the Network Simulation and Gaming Project worked closely with the staff of another EDUCOM activity, the Planning Council on Computing in Education and Research, in developing a procedure to quantitatively compare prices charged and resources utilized for identical services (36).

The procedure developed was based on a set of nine FORTRAN benchmark programs. Within the constraints of the FORTRAN language, this set of programs was designed to represent a variety of computing tasks including small debug runs, "number crunching," and file processing. A brief description of each program is given in Figure V-2. The full set of listings and instructions appears in Appendix VIII.

In order to insure a valid basis for price comparison, the conditions under which the programs were run were carefully specified. Three compiler types were identified (student, standard, and optimizing) and treated separately. Minimum arithmetic precision was specified as 14 decimal digits. Job turnaround time (as a function of priority) was specified such that, on a typical day, there would be at least a 90% likelihood that results would be in the user's hands within an hour after submission.

Figure V-2

The Benchmark Programs

Program Name	Brief Description
TRIVIAL	Does nearly nothing in order to highlight job overhead and minimum charges.
CRUNCHER	A loop containing the four arithmetic operators is executed one million times.
MATMUL1	Two 60 X 60 matrices are multiplied fifty times.
MATMUL2	Two 221 X 221 matrices are multiplied once.
CTOD	Card-to-disk, 2,000 data cards are read and 10,000 card images are written sequentially on disk.
DSKRD	Disk-read, the sequential file created by CTOD is accessed and summarized.
PUREIO	50,000 binary card images are written to disk.
ARMWHIP	Writes and reads a 20-million-character random-access file nonsequentially.
ARMGLIDE	Writes and reads a 20-million-character random-access file sequentially.

Since prices may differ depending on the type of user, three distinct price schedules were defined. Internal prices were defined as those that would be charged to an on-campus research project supported by outside funds. External prices were defined as those charged to an educational user from an un-affiliated university. Wholesale prices were also collected, but there was a wide variation in definition and criteria for these and they are not illustrated in this section.

Figure V-3 presents several internal price statistics for selected program/compiler combinations. Twenty-six installations were surveyed, but in several cases a smaller subset of prices were reported. Hardware, software, and administrative constraints precluded execution of some tasks at some sites. Entries in the bottom row in the table indicate the enormous variation in price for each task -- price differentials with ratios as high as 45 : 1. Since the highest and lowest prices are likely to contain anomalies, another comparison was made eliminating the highest and lowest 15% of the prices reported for each task. Ratios for this comparison are shown on the next-to-last line and still range as high as 9 : 1. More than one installation does not charge at all for jobs such as TRIVIAL, where the overhead necessary to account for resource usage might exceed the resources used by the job. However, it is surprising that one installation had a zero price policy for CRUNCHER, which had an overall average price of $7.57!

The dollar amounts represented in Figure V-3 allow a comparison of individual job prices across installations. However, to make a general comparison of the prices that would be charged for all research computing or all university computing, the composition of the broader workload must be known. Quantitative characterization of a workload is fraught with difficulty. Such comparisons are usually made by taking a sample of actual jobs and running them at other installations. Because this procedure is both costly and time-consuming, it is used only when large purchases are being considered.

The synthetic workload procedure described here provide a basis for comparison with only a moderate effort. There is some loss in accuracy. The procedure defines a workload or job mix as a linear combination of the thirteen tasks cited in the benchmark survey.

The price statistics shown in Figure V-4 are based on a linear combination of the individual jobs shown in Figures V-2 & V-3. This mix is felt to be representative of the academic workload at at least one university. As may be seen in Figure V-4, the workload would cost an on-campus user at the lowest-price facility $60.10. A user with the same workload at the highest-priced facility would pay $338.96. If a user could send the individual jobs of the workload to the facility with the lowest price for each job type, the workload price would be $21.62. Of course, a user sending jobs to several locations would not necessarily pay internal rates and would also incur communications costs.

In evaluating any benchmark survey and comparing prices, many variables must be considered. The choice of tasks and the use of FORTRAN obviously introduce bias. In addition, there are many applications that may not be adequately represented by one or a combination of the benchmarks. For example, the use of packaged software such as SPSS is written in FORTRAN. Since administrative and other file processing applications would be more likely to use PL/1, COBOL, a report generator, or a data base management system, administrative applications are not represented well in this survey. Finally the use of interactive computing is not included in this survey, although it is an increasingly important mode of computing.

Other differences among installations are important but less obvious. Ideally, all costs incurred in providing computing services would be treated in accordance with "generally accepted" management accounting practices. Such treatment would improve the comparability of prices; however, important cost components, such as space and utilities, are not accounted for in a uniform way. At least one installation does not include hardware in its

cost base used for determining prices. And, when hardware is included, the acquisition cost used often reflects substantial vendor discounts that are no longer available.

Federal, state, and institutional policies also frequently impose unusual constraints. For example, a federal policy of disallowing the cost of capital as a legitimate component in setting prices for federal contracts shrinks the cost base when equipment is purchased rather than leased or rented. Explicit subsidies from general university sources, whether these subsidies are planned or the result of unforeseen deficits, also compound cost and pricing problems.

Of the 26 installations surveyed, 20 supplied external as well as internal prices. The external price is defined as the price charged to an educational user not affiliated with the university supplying the service. In a national computer resource sharing network, remote users would ordinarily pay external prices. The most common external pricing strategy is a simple percentage surcharge on internal prices. Figure V-5 shows the surcharge percentage for the 20 installations reporting external prices.

It turns out that the installation with the lowest price shown in Figure V-4 has external rates identical to internal rates. Any educational user would thus pay a price of $60.10 for the workload done at that installation. If a user paid external rates and split up the workload by job type in order to send jobs to the installations with the lowest individual job price, the price would be $23.83 and jobs would be run at ten different installations. Even local users at the installation with the lowest overall price might be interested in the lower individual job prices at other installations.

In a mature network it is likely that external prices will receive more scrutiny from the supply installation and from the user. External buyers will also need to consider other costs. The results shown here do not include any communications costs which, if

Figure V-3

FORTRAN Benchmarks, Internal Price Statistics

	TRIVIAL			CRUNCHER		MATMUL1	
Compiler type	Student	Standard	Optimizing	Student	Standard	Standard	Optimizing
Number of installations with successful run	17	25	18	13	26	24	20
Mean price ($)	0.21	0.55	0.73	7.57	2.99	20.17	13.93
Median price ($)	0.14	0.48	0.63	6.23	2.19	16.48	8.73
Highest price ($)	0.91	1.51	1.90	18.42	10.76	54.19	57.57
85th percentile price ($)	0.37	0.84	1.09	12.16	4.49	39.26	26.74
15th percentile price ($)	0.00	0.15	0.29	3.37	1.15	6.20	3.00
Lowest price ($)	0.00	0.00	0.17	0.00	0.67	3.18	2.05
85th percentile/15th percentile	–	5.52	3.72	3.61	3.92	6.33	8.91
Highest/lowest	–	–	11.18	–	16.06	17.04	28.08

	MATMUL2	CTOD	DSKRD	PURIO	ARMWHIP	ARMGLIDE
Compiler type	Standard	Standard	Standard	Standard	Standard	Standard
Number of installations with successful run	9	24	23	25	20	19
Mean price ($)	73.79	7.37	2.85	17.40	84.13	70.64
Median price ($)	21.43	6.69	2.29	10.31	51.11	49.64
Highest price ($)	414.95	18.04	8.16	59.10	405.46	231.22
85th percentile price ($)	72.73	9.70	3.56	35.39	143.60	149.86
15th percentile price ($)	10.80	4.15	1.44	3.85	16.95	15.87
Lowest price ($)	10.56	3.54	0.98	3.21	10.14	5.04
85th percentile/15th percentile	6.73	2.34	2.47	9.19	8.47	9.44
Highest/lowest	39.29	5.10	8.33	18.41	39.99	45.88

Note: N = 26 installations

Figure V-4

Synthetic Workload Internal Price Statistics by Installation

Price basis	Workload Price
Lowest Price ($)	60.10
15th percentile price ($)	75.10
Median price ($)	127.66
Mean price ($)	144.82
85th percentile price ($)	211.46
Highest price ($)	338.96
85th percentile/15th percentile	2.82
Highest/lowest	5.64

Figure V-5

External user surcharges

Percent surcharge	No. of installations
0	9
1-10	6
11-50	3
>50	2

included, would make shopping for services less attractive. In particular, if a file of twenty million characters, as necessary for ARMWHIP and ARMGLIDE, were shipped over the network, substantial communications costs would be incurred. Remote users would also face additional costs if they needed to support their own user services with, for example, documentation and consulting aids.

E. Observations on Data

1. Overview - Questionnaire I was very general and relatively unstructured, whereas Questionnaire II was very specific in its data requests. The responses to both represented a wide range of quality and quantity. Much of the data required for Questionnaire I could be obtained directly from existing documents, and this approach was used by most of the respondees. Since the questionnaire was circulated before the model design was finalized, the survey was primarily intended to provide an overview of basic institutional facilities, characteristics, and organizational structure. The responses were evaluated for completeness and content, but no attempt was made to force consistency across all institutions. In general the material supplied the project staff with enough information to complete model design and to ensure that the model was capable of representing most of the organizations, facilities, and user populations in existence at the various network sites.

Questionnaire II required perhaps even more effort on the part of the participants because the questions were now very specific and detailed. In many cases the data were not readily available in the desired form, if at all. As with the first questionnaire, there were several instances in which institutions did not have the data requested but came to the conclusion that this was important management information that should be available. Both questionnaires provided some sites with a new viewpoint towards data collection and for others proved to be an impetus towards the organization of data for their own use. At least two

institutions have made some of this material a regular management reporting requirement.

2. General Observations and Problems - As mentioned earlier, the first questionnaire was used primarily to provide the project staff with an overall view of the computer facilities, offerings, policies, and decision-making processes at the participating institutions. Most of the following detailed discussion will deal with the responses to the second questionnaire.

As responses to Questionnaire II were received, they were reviewed for completeness and consistency -- both internal and with other sources of information such as the first questionnaire. Consistency checks were made to ensure that the resource capacities and units used in the various tables were compatible; and that the resource prices, resource requirements, estimates, billing algorithms, and job cost estimates were consistent with the total job cost data. Finally the total resource utilization costs were compared with the total job costs and with the budgeted data. For almost every questionnaire, a number of additional telephone contacts were needed to clarify definitions and procedures and to iron out inconsistencies. It was a large complicated questionnaire, and much work was required to ensure consistency and accuracy.

The first major difficulty encountered was that of the multitude of service types. Many sites were unable to provide detailed data on such fine job categorizations since the information requested was often kept only on an aggregate level (if at all). Initial background work had reduced the number of proposed service types (originally several hundred) to a total of 102, representing 42 major job categories. These included 5 high-priority restricted batch categories such as WATFIV and PLC, 20 general batch categories (each having 4 priorities), and 17 interactive categories. This set of service types was used in the second questionnaire. It quickly became clear that this was still too many -- both from the model perspective and for the information available point

of view. Note that in addition to the list used, it was considered necessary to have several more unassigned service types for unique services. After the responses to the second questionnaire were reviewed, these 102 were further reduced to 44 service types, plus 4 that were reserved for unique services.

The second major problem was that of data consistency. Since it was left to each site to determine the characteristics of each service type at their institution, there was no guarantee of equivalency over all sites. Even within a site the usage of a particular resource by one service type did not always seem to correlate with the usage by another service type.

In addition to inconsistencies in the source data, two problems were encountered in trying to match resource and job costs. The resources listed did not necessarily include all of the resources charged for at each institution, due to the maximum of eight resource types allowed in the questionnaire, and, in some cases, to the complexities of the pricing algorithms. Further, although individual pricing policies for resource use were frequently not linear, the model assumes linearity. In each case the best linear approximation to the resource cost was estimated and used within the simulation model. The effects of these approximations was to cause differences between the listed price and the calculated job costs based on resource use. In most cases the model price was low because of the resources not included. To compensate for these irregularities, a pseudo-resource type called "other" was introduced. The value of "other" was calculated for each service type to bring into balance the listed and calculated job costs. This was discussed earlier under Pricing (Section IV-H).

After all processing steps were completed, the data were converted into machine readable form for use in the simulation model. The approach used is discussed in Appendix III-A in Volume II. Although much effort will still be required during Phase II

to resolve the difficulties, in view of the complexity of the problem and compressed time-scale for this activity the effort has been surprisingly successful.

VI. PHASE I EXPERIMENTS

A. Overview and Purpose

As stated elsewhere in this report, emphasis in this Phase I study was on constructing and validating the basic simulation model. Phase II efforts will focus on capturing the flavor of each participating institution -- its policies, practices, and decision behavior. Much also remains to be done in the way of "tuning" the hardware and workload representations so that model outputs truly reflect reality (or at least what institutional representatives perceive to be reality). Clearly, many of the more interesting experiments with the model must await the completion of these tasks. It is not yet possible to make definitive statements about the implications of particular policies on specific sites, or on the impact of network membership on any real site.

However, even with the above limitations, there are a number of useful things that can be done with the model. It is fully operational, can handle any particular combination of policies specified, and has on-line data files representing a number of sites. Although the data files require additional validation and "tuning" by the actual institutions, they do contain complete and consistent sets of data and policies that could represent a possible site.

This chapter describes a number of experiments that have been or could be completed with the model in its present form. As the design of the model progressed, a fairly comprehensive set of desirable experiments was specified (Section VI-B). These are all areas that are critical to the understanding of networks and the likely impacts of networks on member institutions. Although a detailed experimental procedure was developed for each experiment (Sections VI-C and D), only a subset of the list has been completed. There were several reasons for designing a more comprehensive list

of experiments than was reasonable to complete. First, the list provides an indication of the capabilities and limitations of the model. This was a useful exercise for project personnel in that it forced them to state rather explicitly just what goals were set for the project, how these goals would be achieved, and how the model could be used to provide the quantitative data. For non-project readers, it gives a good indication of the types of results to be expected and the level of "user" towards which project efforts and model outputs are directed. Perhaps the greatest value derived from detailing a large set of possible experiments was that this process verified that the existing model was capable of handling them.

B. Areas of Experimental Interest

The selection of experiments appropriate for the simulation model was based upon a consideration of the questions about network operation that seemed to be of major importance. After discussions among the various members of the project team, agreement was reached on 11 areas of particular interest:

1. Standard Performance
2. Bilateral Agreements vs. Central Network Organization
3. Site Specialization
4. Network Stability
5. Network Resource Sharing Potential
6. Communications Costs
7. Service Pricing Policies
8. Provision of Special Services
9. Network Equilibrium Conditions
10. Quality of Network Information Made Available to Users
11. Network Growth Effects

C. Conduct of Experiments

After the primary areas of concern were identified, attention was directed toward the specification of appropriate experiments within each of these areas. Most of the experiments were limited to 22 time periods and were run with five sites as described in Section D-1. The number of sites and time periods was arbitrary for

experimental purposes, and more could have been used. These choices were based on a desire to minimize run costs at this time. As the experimental results indicate, there were considerable network flows even with only five sites and less than six months of simulated time.

Each experiment was conducted by changing the appropriate data files for each of the five sites to reflect the policy areas or supply offerings under investigation. Occasionally, a separate subroutine was written to model unique situations. In the shock experiment, for example, a modification to routine XOGEN (3.1) was written, which made site 3 unavailable as a supplier in period 10 and available once again in period 11. This was accomplished with nine lines of FORTRAN code and demonstrated the flexibility of the model and the ease with which such additions can be made.

Although it is expected that in a real network the stabilization of network flows will take a long time, it was possible in the experiments to select policies that hastened this process so that most shifting had been completed by week 10. Several of the experiments required that perturbations be made to a stable network, so these runs were carried out over a period of 22 weeks (with the perturbations occuring in week 10) to allow observation of the effects of these perturbations.

D. Experimental Runs

1. Standard Performance

a. Experiment - The major intent of this experiment was to determine a "base" performance level to serve as a standard of comparison for the performance data derived from subsequent experimental runs.

The standard run on which comparisons were based reflected an idealized environment in which all work could be done (if desired) at any network supplier. Hardware incompatibilities, conversion costs, delays, and network surcharges were ignored. The common preference to do work "in-house" if possible, was reflected by giving each site a 20% internal rating boost as compared with outside suppliers. Similarly, the problems involved in switching suppliers were represented by a corresponding rating boost in favor of the present supplier. In order to better examine the potential for network usage, it was assumed that none of the sites put artificial restrictions on where or how users spend available funds.

b. Results - The configuration of the five sites used in this and the following simulation studies were as follows:

Site	Description
Site-1	IBM 370/158 medium load
Site-2	IBM 370/145 heavily loaded
Site-3	IBM 370/168 lightly loaded
Site-4	No internal hardware. Only interactive demand which is currently satisfied externally from a non-network site.
Site-5	H6180 - large timesharing system. Extremely I/O bound. Specialized user community that does not have any batch usage.

In the base run, sites 1 and 2 quickly became heavy network users; site 3 received a large amount of work from the network (both batch and interactive); site 4 shifted approximately 10% of its demand to the network; while site 5 (after raising I/O charges) sent much of its I/O bound interactive work to site

3 and received a large amount of interactive work from other users on the network. Figure VI-1 shows the cumulative dollar flows at period 20. Of the total $4,028,694 spent by the users at the different sites, almost 13% was spent on the network. It should be noted that site 2 shifted almost 70% of its total expenditures onto the network.

Figure VI-1

Cumulative Dollar Flows - Standard Run (20 weeks)

	Total User Expenses	Network Expenses	Network Income	Balance of Trade
Site 1	$ 545,761	$168,444	$ 57,539	$-110,905
Site 2	400,662	271,937	1,484	-270,453
Site 3	1,076,609	9,079	314,993	305,914
Site 4	332,154	33,948	0	- 33,948
Site 5	1,673,508	29,690	138,082	109,392
	$4,028,694	$513,098	$513,098	$ 0

2. Bilateral Agreements vs. Central Network Organization

a. Experiment - This study investigated some of the effects of a central organization which would facilitate usage of multiple sites by a remote user. For example, such a central organization might provide account numbers for multiple-facility use, central billing services, and standardization (or automatic translation) of job control set-ups and procedures at each site. The alternative to having such a central organization would be to rely upon a series of bilateral agreements between various institutions on the network. It was felt that the central organization would facilitate use of multiple sites by users but that the central organization might also require some type of surcharge in order to support the services it would provide.

The positive effects of a mature multilateral network were represented by setting DEBUMP, the in-house rating increase factor, to 1.0 rather than 1.2 to reduce the favoring of in-house usage; ICDP, the momentum factor, was set to a low value so that demanders did not have a great allegiance to the site where they were currently doing their computing. As an initial experiment, no surcharges were imposed.

b. Results - The most noticeable behavior observed in these experiments was the large percentage of interactive work on the network. Within 22 weeks, 33% of all interactive work was being satisfied on the network. In addition to the interactive usage, 16% of the batch jobs were sent over the network. This was a surprisingly high percentage considering the communications costs associated with batch network work. As might be expected, most of the batch flows were directed to site 3, the 370/168. By week 20, site 3 was receiving almost as much income from network users as it was from its inside users. Site 5 heavily loaded at the beginning of the run, quickly became completely saturated. As the prices of its overloaded I/O channel were increased and turnaround became intolerable, many of its I/O intensive users moved to site 3 to avoid the higher prices. Over a longer time period, either the I/O capacity at site 5 would be increased to accommodate the demand or much of the usage would have to be discouraged.

Because of the present lack of understanding of factors influencing shifts of workload (policies, user behavior, etc.), it is dangerous to put much credence in the above results other than to recognize that, given present price and service levels, a significant percentage of users might be willing to go somewhere else.

Similar experiments with higher communications costs indicate that a nominal surcharge for the central facilitating function would not be a major factor in network usage. (Experiments should be run to determine the point where surcharges would begin to affect flows).

3. Site Specialization

a. Experiment - The primary intent of this area was to investigate the effects of site specialization in such services as low price, user support, or rapid turnaround. The questions to be answered concerned the viability of centers that try to specialize in this fashion, and the types of user response to these services. For instance, if a site emphasized those service types that were efficient on its particular hardware, it could charge less for those service types than a site with a comparable hardware configuration but which offered a full range of services. By specializing in the services it offers, a site can also focus on the hardware configuration necessary to support those services.

One implication of site specialization is the shifting of usage of some service types to other network sites that process them comparatively more efficiently. Just as in an international trade situation, where different countries have different mixes of raw materials available to them, and hence trade between them becomes beneficial, so in a networking situation one would anticipate certain service types to flow to particular sites which have a comparative advantage in that type of computing. Thus in a networking situation, one might expect that the distribution of service types processed at a site would change over time, as users moved away from that site and onto the

network for selected service types. At the same time, users from the network should, on balance, be expected to become users at that site for the services which it did provide efficiently. If this phenomenon occurred, more work could be done on the network than could be done at the sites individually -- without any increase in resources!

b. Results - Although separate experimental runs were not made in this area, observations on site specialization were made in a number of the other experiments. Every installation had some users going onto the network for at least part of their needs, and every site that offered services found some outside buyers for some of their services.

Site specialization depends to a great extent on user behavior and policy decisions. As these aspects are developed more in Phase II and a wide selection of policies become available, this topic will be more thoroughly investigated.

4. Network Stability

a. Experiment - The major questions to be answered in the area of network stability concern the conditions under which institutional behavior will become unstable and lead to wild swings in use of the network. Typical factors that might lead to this condition include:

- lowered barriers to switching
- price competitiveness among network sites
- shifts in behavior patterns
- capacity shocks

Lowered barriers to the switching of suppliers will evolve from technological advances (easier technical access) and a central facilitating organization. Various levels can easily be represented in the model

by adjusting the momentum parameter which governs the tendency of users to switch sites. As this value is decreased, users will find it easier to switch sites. Thus, when difficulties occur at one site, there will be a tendency to have a mass exodus of users to other sites. This switch could, in itself, cause problems at the new supplier sites (i.e., overloading) which would encourage even more movement. A number of experiments could be conducted examining network behavior as this momentum parameter is varied. Special circumstances could induce instability even in normal circumstances. For example, an artificial shock which temporarily induced poor turnaround at several sites might trigger such a reaction.

A variety of price cutting situations could also cause abnormal network flows. Suppose, for example, an under-utilized site is successful in attracting work by virtue of a series of price reductions. What if varying numbers of other sites respond in kind? What if still other sites join in the competition? What if selected larger sites act in a predatory manner?

In the behavior area, shocks introduced by sites shifting to entrepreneurial behavior, to zero net balancing behavior, or to cost minimization behavior could create instability within the network. The number of sites and possible behavior patterns allow a wide range of experimentation in this area.

A network in equilibrium could be vulnerable to a variety of perturbations in available capacity. Typical examples might include the addition of a new network supplier offering a substantial portion of network capacity at reduced prices; the addition of a new network demand source having a demand equal to a signifi-

site 1, which still had some excess capacity. All of their work was accommodated, albeit somewhat slower.

No oscillatory behavior developed with this particular shock. Most of the users returned to site 3 rather quickly since, on balance, its offerings were significantly more attractive than available alternatives.

A similar experiment was to unexpectedly reduce the supply at site 3 to a small percentage of its normal value, thereby simulating a week of excessive, unanticipated down-time. In this case, users still attempted to go there, but there was a large amount of unsatisfied demand. In addition, site 3 turnarounds became intolerably high. As one would expect, in the weeks following the shock, many turnaround-sensitive users moved from site 3 to other sites on the network.

These simulation runs indicate that in this representation, the network is quite stable. There are a number of factors that tend to dampen movement. These include communications limits, increased turnaround times, and site capacities. After site policies and decision-making rules have been further refined, the area of network stability can be explored in more detail.

5. Network Resource Sharing Potential

a. Experiment - A major question concerns the degree of sharing which can take place under various environmental conditions. The types of conditions that could be tested include:

- Situations in which there is a wide spread in the service type costs offered by the different facilities, providing users with economic incentives to share resources.

- Situations in which there is a wide spread in turnaround time between service types and between different sites on the network, providing the user with a response incentive to share resources.

- Situations in which different sites specialize in different types of services, giving the user a service incentive to share resources.

- Situations in which the external price of services is perceived by all users to be less on the network than it is at their local site, providing another type of economic incentive to move onto the network.

- Situations in which there are no communications costs, reducing an economic barrier to sharing.

b. Results - All of the above conditions were tested during the simulation runs. Most of the results were as expected and will not be detailed here. In general, users moved from one site to another based on major imbalances in any of the following areas: price, turnaround, or support.

c. Communications Costs

a. Experiment - In considering a national network, one of the prime questions to be answered concerns the point at which communications cost will begin to affect the volume of network traffic. It is also important to determine the types of users or services which would feel the impact first.

b. Results - The effect of communications costs was studied by varying communications costs over a range from zero to $50 per kilo-packet. (In certain situations, current prices are approximately $.60/kilo-packet).

Surprisingly, there were reasonable flows even for high communications costs. In one run, with communi-

cations costs set at $3/kilo-packet, 14% of the batch work and 21% of the interactive work was done on the network. Apparently, the high price differentials that exist between sites can outweigh seemingly high communication costs. When communications costs are reduced to very low values, network usage increases even more, as an increasing number of service types become less expensive at other sites.

It should also be noted that communications costs dramatically impacted the type of work done over the network. When costs were high network usage was primarily devoted to service types that generated relatively little network input or output. As communications charges dropped, more communications-intensive service types became economical to do over the network. As an example, when communication costs were in the range of $10/kilo-packet, site 1 only sent jobs from three different service types to site 2. When costs had dropped to $.05/kilo-packet, site 1 sent 21 different services types to site 2. Flows in the reverse direction went from 10 different service types to 21 different service types as communications costs were lowered.

Lowering communications costs appears to broaden the types of work done via the network. Most of the increased flows appear to be from the increasing number of different service types for which there are network flows rather than from increases in already-existing service types due to direct price impact.

7. Service Pricing Policies

a. Experiment - In a cooperative network composed of independent institutions, there are important questions concerning the impact of sites using different types of pricing strategies. For example, what is the

impact of a facility that seeks to be competitive, that seeks to base prices only on cost recovery, or that seeks to base prices on cost recovery plus profit? Most of the data on network behavior under different pricing conditions can be obtained from the runs associated with Network Stability (relating to price and capacity) and those on Network Equilibrium Conditions (relating to entrepreneurial behavior).

b. Results - The initial experimental results on pricing policies focuses on policies in which sites priced in order to maximize resource utilization. Sites 1, 3, and 5 all followed an automatic pricing policy that lowered resource prices on resources that were (greatly) under-utilized, and raised prices on resources that were (greatly) over-utilized. Considering only two resources, CPU and I/O, site 5 tripled its I/O prices during a typical run, while site 3 lowered its I/O prices by approximately 25%. These resource price changes for these two resources affected the per-unit price of all service types. Of course, the price changes were reflected more directly to users who use service types that were intensive in the use of the resource in question. At both sites the policies had the desired effect. I/O intensive jobs migrated away from site 5, thus enabling the facility to increase its total number of jobs. Similarly, site 3 attracted much new work to its facility due to its price reduction.

8. Provision of Special Services - Possible Experiments - In the area of special services, there are a number of questions to which answers are sought, including the ability to reward entrepreneurial risk (investment in the development of such services) through surcharges on the use of these special services. If the network is to have a means of encouraging the development and support of new services and specialized software, it may be necessary to

have some sort of surcharge or royalty arrangement. Thus far, the absence of reasonable estimates of demand or response patterns for new services has limited experiments in this area. This type of experiment is particularly well suited to the gaming phase of the project in which users can react directly to such offerings.

One useful set of experiments might assume several reasonable (possible) demand patterns. For each pattern, runs could be made under the assumption that there is no impact upon demand of various levels of surcharge. Hence, the unhindered growth pattern of the demand for a new service could be examined over time. From this could be calculated the revenue pattern over time that would result at various levels of royalty. This revenue pattern would be an optimistic one, but would serve as basis for comparison with development and support costs for these services. Following this, a set of runs should be made to look at the impact of various levels and types of surcharges on demand. Thus runs might be made with a range of surcharges, starting near zero and continuing until a surcharge was reached that would choke off demand faster than revenue would increase.

9. Network Equilibrium Conditions - Possible Experiments - An important set of questions to be answered about the network concerns the characteristics that the network might have when it reaches a stable equilibrium condition. The particular types of site behavior patterns to consider might include:

- Entrepreneurial - The computer operation as an "empire builder."
- Zero net balance of trade - The expenditures of the facility's local customers on outside services is to be approximately equal to the expenditures of outside network users at the local facility.
- No capacity increase - The computer facility does not increase its processing capability when full capacity is reached.
- Delayed capacity increase - The computer facility adds services or capacity only after the additional demand has

been "assured," leading to a lag between the reaching of capacity operations and the acquisition of additional processing capability.

- Cost minimization - Users are encouraged to take advantage of the most economical sources of supply over the network (including in-house). The facility reduces capacity if necessary in order to match its supply of services with local demand.

- Site specialization - The facility specializes in some particular capability such as low-priced service, rapid turnaround, or user support.

The basic experimental pattern would be a set of simulation runs that would test varying numbers of network sites exhibiting the particular behavior. Thus runs might be planned with one site, 50% of the sites, and all of the sites exhibiting entrepreneurial behavior. Similarly, a set of runs could be made with a varying number of sites exhibiting no capacity increase behavior, zero net balance behavior, and cost minimization behavior. In the case of delayed capacity increase behavior, two sets of runs would be required. The first set of runs would involve a varying number of sites exhibiting moderate lags between the time capacity is reached and the time additional processing capability is installed, while the second set would involve a varying number of sites having abnormally long lags between the time capacity is reached and the time additional processing capability is installed.

Site specialization experiments present a more complex problem. First, a series of runs should be made with only one site specializing in low price, rapid turnaround, user support, or a special service type. Then a set of runs can be made with a varying number of sites (e.g., 50%, 100%) having the same specialty. If the runs made with a single specializing site indicated different behaviors or different equilibria being reached as a function of the type of specialization (e.g., low price), then it will be necessary to replicate the runs (with varying numbers of specializing sites) for each of the different types of specialties. Following the examination of sites having the same specialty, it will be necessary to experiment with varying

numbers of sites having different specialties to see if a balance in types of specialties offered has any impact on the equilibrium reached by the network.

a. Experimental Results - Conduct of the full experiment would involve a large number of simulation runs. However, many of the specific questions can be examined by analyzing existing data from other experiments.

1) Empire Builder - In most of the simulation experiments run, site 3 has many of the characteristics of an empire builder. It is lowering its prices to try to build up demand. It has high support for most service types, and it clearly has much more capacity than it needs. This behavior was rewarded in all of the simulation experiments. Site 3 consistently showed a net surplus from its network usage, and it quickly took over a significant amount of the interactive work on the network. Future experiments will have to look more carefully at the situation in which there are multiple sites exhibiting this behavior pattern.

2) Zero Net Balance - Several runs were made to investigate site behavior under zero net balance of trade policies. The results were largely a function of the composition of the network. If the network contained a large percentage of demand-only sites, for example, then zero net balance of trade restrictions at particular sites had little effect on aggregate network flows. On the other hand, if all sites on the network were suppliers following a zero net balance of trade policy, growth was slow and oscillatory. Some sites with little to offer eventually withdrew from network participation. The general conclusion is that sites with desirable service

offerings (even if only a few) are able to follow this policy with little difficulty. Most others are successful only if there exists a reasonable number of sites that are net demanders of services.

10. Quality of Network Information Made Available to Users -

The major questions in this area concern the effects that the quality of price, turnaround, and support information can have on user behavior and how the behavioral changes relate to the expense of providing various qualities of status information.

The necessary information can be derived from a set of experimental runs with varying types of network performance information provided to users. These include perfect information, perfect information with time lags, and information having varying degrees of randomized distortion as well as bias.

11. Network Growth Effects - A very interesting area concerns the potential that the network has for growth by net service demanders joining the network. Of particular concern are the effects that such additional demanders can have upon the suppliers and upon the behavior of the individual network users.

Much of the data desired in this area will be obtained from the capacity experiments conducted as a part of the network stability tests discussed earlier. However, it would be desirable to make a limited number of additional runs with very large net increases in network demand (e.g., 50%, 100%). These runs should provide an indication of what might result if a number of small non-supplier sites (or a larger site with inadequate internal capacity) were to join the network. They would also provide an indication of the way in which the perturbations caused by the increased demand would work themselves out.

L. Summary of Findings

As stated, the experiments thus far are of a very preliminary nature and it is dangerous to draw many hard conclusions before a full set of institutional behaviors and policies can be incorporated. However, a number of interesting trends were demonstrated that bear further investigation. The experimental results have the following implications for the viability of an actual institutional network:

1. Network Flows - Substantial forces exist encouraging network flows. There are large discrepancies among sites on the network in the area of prices, turnaround, and user support. Assuming that site hardware costs are not increased or decreased (in the short run), one would expect the average price per job to decrease significantly in a network environment. In addition, average job turnaround would decrease, and average support levels per job would increase. The major inhibition to network flows would be policies through which sites attempt to prevent their users from using the network because of a cash flow drain. This deterrent would be greatly reduced if there were a reasonable number of sites that were net-demanders in-balance; so that all supplying sites could attract income to help support their network usage.

2. Network Stability - None of the simulated runs thus far exhibited unstable behavior even in the presence of relatively large shocks. There seem to be a number of stablizing factors which contribute to dampen oscillatory movement and the implication is that the network is quite stable in its current representation.

3. Communications Costs - Communications costs do not appear to be a significant deterrent to network usage due to the large discrepancy among sites' current prices. However, it is unlikely that price differentials as large as those that presently exist would remain in a real network environment. Hence, flows would be less than reported here and the sensitivity to communications costs would be correspondingly larger.

4. Central Facilitating Network - It appears that the potential exists for a positive role for a central facilitating organization. A facilitating organization should encourage efficient use of the network, and should have some method of funding. The simulations indicate that both of these conditions appear to be true. First, the multilateral simulations showed considerably more flows than the bilateral runs; and, second, the communications costs experiments indicated that a surcharge on network traffic could be used to finance a central facilitating organization.

As the policies and site behavior are enriched in Phase II, the wide selection that will be available will allow even further experiments. Some of those that have already been done can be expanded and examined in more detail and those that were only outlined here can be performed. Thus, through such a set of experiments more can be learned about the implication of various policies and behavior patterns on both the participating institutions and the network.

VII. SUMMARY OF PROJECT STATUS

The purpose of this section is to provide an abbreviated description of the project, its current status, and plans for Phases II and III.

A. Simulation Model

1. Characteristics

- Is designed with a modular structure in a top-down fashion.
- Permits relatively easy (often trivially so) insertion of new modules and modifications to existing modules.
- Is written in FORTRAN for maximum transportability.
- Follows well-defined conventions to increase readability, avoid errors, and simplify maintenance.
- Operates with a weekly time interval.
- Incorporates a wide range of decision variables that permit the exploration of policy alternatives.
- Defines computing services in terms of (currently) 44 different standard "service types" which are relatively homogeneous services, such as "debugging runs" and "short statistical packages."
- Permits each site to define (currently) four special service types that are unique to that site and of general interest to outside users.
- Defines capacity at each network site in terms of seven standard basic resources such as CPU speed and primary memory size and one optional resource which may be selected by the site.
- Estimates for each time increment, the demand at each site for each service type.
- Maps service types into requirements for each resource.
- Determines actual demand at a site by aggregating demands from all sites (including local demand) and then accounts for constraints on cash flows, communication capacity, etc.

2. Status

- Is currently operational.
- Contains approximately 15,000 source statements (1/3 executable statements, 1/3 non-executable statements, e.g., data definitions, 1/3 comments).
- Is thoroughly documented.
- Has been validated and is capable of performing all specified analyses.
- Requires expansion of the library of available policies and practices -- minor task continuing throughout the project.
- Requires a major effort in additional validation of site data which is often incomplete or inconsistent.

B. Site Data

- Collection of data has proven to be a problem because of differences that exist among sites with respect to data that are routinely collected, classification of services, classification of resources, costing conventions, etc.
- Collection of data was accomplished with two questionnaires: the first one was unstructured to gain a better overall understanding of each site, and the second one was structured according to model requirements (particularly with respect to current service type supply and demand).
- Benchmark programs have been run and are available to verify data across sites.
- Initial data are available on all 16 sites and steps are being taken to collect missing data and resolve inconsistencies.
- Data from seven sites are considered complete enough for valid preliminary network studies.

C. Status and Implementation of Background Studies

1. Service Types and Workload Representation - Most of the background studies have been completed and the results have been incorporated into the model. For these, the resultant model char-

acteristics are described. For studies still in progress, the current status is presented.

- "Service types" are used to characterize relatively homogeneous (with respect to machine impact) computing services.
- Dimensions considered include batch versus interactive jobs, type of resources required (e.g., CPU or input/output), size of job, and priority.
- It is desirable to keep the number of service types below 50, since this number is the major determinant of the primary memory requirement for the model. It also represents an upper (perhaps too high) limit on the level of detail at which sites are able to describe current activities.
- Representation currently consists of 48 service types: 11 interactive with priority 1, 1 fast batch with priority 2, 32 general batch with priorities 3 to 6, and 4 available for unique services.
- Demand in terms of each service type is mapped into resource requirements at each site.
- Seven resource types are used, with a site dependent eighth resource type permitted.

2. Network Organization and Administration

- A variety of alternative network organizational and administrative structures have been defined.
- Any combination of these structures can be represented by proper specification of existing model parameters.
- Policy variation encompassing a wide range of alternatives is permitted.

3. User Services and Support

- Level of service is expressed in terms of dollars expended.
- Budget for user services at a given site is allocated across each service type. A variety of allocation schemes is permitted.
- Demand for a service type depends (in addition to price and turnaround) on expenditures for user support of that service type.

- Cost of user services includes both fixed and variable costs.

4. Communications

- Representation is capable of accommodating a wide variety of technologies and price structures.
- Model currently utilizes communication technology and prices as provided by commercial organizations such as Telenet.
- The cost of communications includes an initial interface cost, which is dependent on type of interface (e.g. modify host operating systems versus dial-in public port).
- Costs include a fixed period charge, which depends on bandwidth and distance to nearest entry node, a variable packet charge.

5. Computer Systems Performance Modeling

- This has been one of the most difficult technical problems faced in developing the model.
- Problems stemmed from heterogeneity of hardware at the different sites.
- Current approach taken (in Questionnaire #2) is to ask each site for volume and performance data by standard service types and to utilize this data directly in simple tabular form.
- A parallel study has been made to apply network queueing models to the performance measurement problem in an attempt to obtain analytical representations. Early results are very encouraging.

6. Supply Determination

- An institution's supply policies pertain to budget hardware, software, service offerings, support services, and prices.
- Level for each element of supply (e.g., hardware) is governed by the budgeted funds allocated to that element, the actual or perceived demands for it, and site policy decisions.

7. Demand Estimations

- Demand at a given site is initially estimated by user category as a function of past usage, growth trends, seasonal effects, turnaround, price, and user support.
- Aggregate demand at a site by user category is then mapped into demand by individual service type.
- Service type demand at a user site is then allocated among competing supplying sites based on price, turnaround, user support, and inertia (i.e., reluctance or inability of users to make rapid shifts in their demand among sites).

8. Pricing

- Sites can set prices for each service type directly.
- Most sites prefer to set prices for raw resources and let the model calculate service type prices.
- Typical of the pricing strategies available to supplier sites is that of raising prices of heavily utilized resources and lowering those of under-utilized ones -- thus encouraging more efficient overall system utilization.

9. Site Representations

- Capacity at a site is defined in terms of the maximum usable capacity of critical resources.
- Each available service type at a site is defined in terms of its requirements for these critical resources.
- Communication capacities, reliability estimates, and scheduled availability are also collected by site.
- Each site presents a unique set of from one to ten user categories (e.g., administrators, students, researchers, etc.), from which demand by service type is generated.
- Policy variables for each site define its demand policies, supply policies, and "market" policies (i.e., how capacity is rationed if all of the demand cannot be satisfied).

- Other site-specific data pertain to such matters as price discounts for specific buyers and non-standard communication costs to specific sites.

D. Phase I Simulation Experiments

- Emphasis was on exercising and validating the basic simulation model.
- A comprehensive set of experiments was designed in detail. These included explorations of such areas as:
  - Bilateral agreements versus central network organization.
  - Site specialization
  - Pricing levels
  - Propensity of users to shift sites in response to lower prices or better service
  - Effects of quality of network information disseminated to users on network usage
  - Communication costs
  - Identification of constraints on resource sharing
  - Dynamic network behavior when major perturbations were introduced
  - Consequences of alternative network structures and arrangements on:

    Supply and demand at each site
    Work flow patterns
    Balance of payment patterns
    Growth patterns
    Equilibrium conditions
- The model is capable of examining all of the situations listed above.
- Several of the experiments which did not require Phase II policy information or a full set of validated site representations were conducted.
- Many of the remaining experiments will be conducted during Phase II as site data and policy representations permit.

E. Perspective Relative to Work in Phases II and III

- Not all basic Phase I site data are available. The rest will have to be collected and validated in parallel wtih Phase II.
- Phase II data collection will include both written questionnaires and on-site interviews. Focus will be on institutional policy and decision making behavior.
- The current model is fully adaptable to site-specific policy and behavioral data.
- Phase II will use the model to represent the actual policies and decisions of the participating institutions.
- Phase II experiments will examine the implications of a variety of site and network policies and decisions on network flows, individual network members, and overall network viability.
- Adoption of the model to allow interactive gaming decisions in Phase III should be less difficult than expected.
- Phase III will allow interactive selection and modification of policies by institutional representatives in order to explore the dynamic aspects of network behavior.

VIII. REFERENCES

1. Report to the National Science Foundation on the Study to Develop a Research Plan for a Simulation and Gaming Project for Inter-Institutional Computer Networking, Grant #GJ-41429 (EDUCOM, Princeton, New Jersey) August 15, 1974.

2. Emery, J. C., "Implementation of a Facilitating Network," Policies, Strategies and Plans for Computing in Higher Education, (EDUCOM, Princeton, New Jersey) 1976, pp. 25-43.

3. Greenberger, M., and J. Aronofsky, J. L. McKenney, and W. F. Massy, editors, Networks for Research and Education: Sharing of Computer and Information Resources Nationwide, (MIT Press, Cambridge, Massachusetts) 1974.

4. "A Simulation and Gaming Project for Inter-Institutional Computer Networking," submitted to the National Science Foundation, Grant #DCR75-03634 (EDUCOM, Princeton, New Jersey) September 1974.

5. "Request for Research Grant Continuation for the Simulation and Gaming Project for Inter-Institutional Computer Networking," submitted to the National Science Foundation, Grant #MCS75-03634 (EDUCOM, Princeton, New Jersey) January 1976.

6. "Planning Council on Computing in Education and Research." (EDUCOM, Princeton, New Jersey) May 1976.

7. Segal, R., and White, N., "Management of a Large Computer Network Simulation Project," (Fourth Annual Symposium on the Simulation of Computer Systems, Boulder, Colorado) August 1976.

8. Baker, F. T., "Chief Programmer Team Management of Production Programming," IBM Systems Journal, 11, 1, 1972, pp. 56-73.

9. Baker, F., and Mills, H. D., "Chief Programmer Teams," Data mation, 19, 12, December 1973, pp. 58-61.

10. Boehm, B. W., "Overview" in "Structured Programming: A Quantitative Assessment," IEEE Computer, 8, 6, June 1975, pp. 38-40.

11. Stevens, W. P., Meyers, G. J. and Constantine, L. L., "Structured Design," IBM Systems Journal, 13, 2, 1974, pp. 115-139.

12. Dahl, O. J., Dijkstra, E. W., and Hoare, C.A.R., Structured Programming (Academic Press, London, England) 1972, 220 pp.

13. Dijkstra, E. W., "Some Meditations on Advanced Programming," Proc. IFIP Congress 1962 (North Holland Publ. Co., Amsterdam The Netherlands) 1962, pp. 535-538.

14. Miller, D. F., and Lindamood, G. E., "Structured Programming: Top-Down Approach," Datamation, 19, 12, December 1973, pp. 55-57.

15. Mills, H. D., "Top Down Programming in Large Systems," Debugging Techniques in Large Systems (Prentice-Hall, New Jersey) 1971.

16. Myers, G. J., Reliable Software Through Composite Design (Mason and Charter Publishers, Inc., New York) 1975.

17. Naughton, J., McGowan, C. and Horowitz, E., "Structured Programming: Concepts and Definitions," IEEE Computer, 8, 6, pp. 23-37.

18. Parnas, D. L., "On the Criteria to be Used in Decomposing Systems into Modules," Comm. ACM, 15, 12, 1972.

19. Weinberg, G. M., The Psychology of Computer Programming, (Van Nostrand, New York) 1971.

20. HIPO Hierarchical Input Process-Output Documentation Techniques, Form No. SR 20-9413, IBM. Corp.

21. Segal, R., and White, N., "Representation of Workloads in a Network Environment," Proc. of the Summer Computer Simulation Conference, Washington, D.C., July 1976.

22. "Computer System Performance Modeling for the EDUCOM Network Simulation and Gaming Project." An informal report to EDUCOM documenting a background study, Stanford Research Institute (EDUCOM, Princeton, New Jersey) June 1976.

23. Buzen, J. P., and Shum, A. W.-C., "Structured Considerations for Computer System Models," Proc. of the Eighth Annual Princeton Conference on Information Sciences and Systems, Princeton, New Jersey, March 1974, pp. 335-339.

24. Gordon, W. J., and Newell, G. F., "Closed Queueing Systems with Exponential Servers," Oper. Res., 15, 2, April 1967, pp. 254-265.

25. Chiu, W., Dumont D., and Wood, R., "Performance Analysis of a Multiprogrammed Computer System," IBM Systems Journal, May 1975, pp. 263-271.

26. Buzen, J. P., "Analysis of System Bottlenecks Using a Queueing Network Model," Proc. ACM-SIGSOPS Workshop on System Performance Evaluation, Cambridge, Massachusetts, April 1971, pp. 82-103.

27. Buzen, J. P., "Computational Algorithms for Closed Queueing Networks with Exponential Servers," CACM 16, 9, September 1973, pp. 527-531.

28. Kobayashi, H., "Some Recent Progress in Analytical Studies of System Performance," Proc. of the First USA-Japan Computer Conference, Tokyo, Japan, 1972, pp. 130.

29. Buzen, J. P., "Fundamental Laws of Computer System Performance," International Symposium on Computer Performance Evaluation, March 1976.

30. Baskett, F. and Palacios, F. G., "Processor Sharing in a Central Server Queueing Model of Multiprogramming with Applications," Proc. Sixth Annual Princeton Conference of Information Sciences and Systems, March 1972, pp. 598-602.

31. Baskett, F., Chandy, K. M., Muntz, R. R., and Palacios, F. G., "Open, Closed and Mixed Networks of Queues with Different Classes of Customers," ACMII.2, April 1975, pp. 248-260.

32. Buzen, J. P., and Goldberg, R. S., "Guidelines for the Use of Infinite Source Queueing Models in the Analysis of Computer System Performance," AFIPS Conference Proc., Vol. 43 (AFIPS Press, Montvale, New Jersey) May 1974.

33. Buzen, J. P., "Cost Effective Analytical Tools for Computer Performance Evaluation," Proc. COMPCON 75 (Eleventh Annual IEEE Computer Society Conference), Washington, D.C., September 1975.

34. Boyse, J. W., and Warn, D. R., "A Straight-forward Model for Computer Performance Prediction," Comp. Surveys, 7, 2, June 1975, pp. 73-93.

35. "A Cyclic Queueing Model for the EDUCOM Network Simulation and Gaming Project," Final Report, BGS Incorporated (EDUCOM, Princeton, New Jersey) in preparation.

36. Heller, P., "Benchmarking the Price of Computing: Results of a Survey," Computer Networks, Vol. 1-1, June 1976, pp. 27-32.

Publications

The following references were supported whole or in part by this project:

Emery, J. C., "Implementation of a Facilitating Network," Policies, Stragegies and Plans for Computing in Higher Education, (EDUCOM, Princeton, New Jersey) 1976, pp. 25-43.

Heller, P., "Benchmarking the Price of Computing: Results of a Survey," Computer Networks, Vol. 1-1, June 1976, pp. 27-32.

Segal, R., and White, N., "Management of a Large Computer Network Simulation Project," (Fourth Annual Symposium on the Simulation of Computer Systems, Boulder, Colorado) August 1976.

Segal, R., and White, N., "Representation of Workloads in a Network Environment," Proc. of the Summer Computer Simulation Conference, Washington, D.C., July 1976.

"Computer System Performance Modeling for the EDUCOM Network Simulation and Gaming Project." An informal report to EDUCOM documenting a background study, Stanford Research Institute (EDUCOM, Princeton, New Jersey) June 1976.

"A Cyclic Queueing Model for the EDUCOM Network Simulation and Gaming Project," Final Report, BGS Incorporated (EDUCOM, Princeton, New Jersey) in preparation.

Appendix I

Model Overview and Flows

The major segments of the model are discussed in detail in this appendix. As described in Section III, the model design is based on a modified top-down, structured programming approach. This discussion follows the sequence specified by the system flowcharts which appear as Figure A.I-1a - A.I-1q following the text. The flow of control in all cases is from top to bottom, left to right. Looping is indicated by cross-hatching as, for example, in Module 3.0 of Figure A.I-1a. The symbol ∀ is read: "for every."

A. NETSIM

Module NETSIM is the main program and serves to control the entire Network Simulation Model. All runs are initiated by calling this module. It, in turn, calls other modules as required. NETSIM performs three major functions:

1. Common variables are initialized to zero.
2. Input/Output unit numbers are set.
3. The five top-level functional routines are called in the sequence indicated in Figure A.I-1a.

Figure A.I-1a summarizes overall model operation, as well as illustrating the modular approach to the model design. Note that the user only interfaces with one module, NETSIM. NETSIM, in turn, does some preliminary initialization. Its major function, however, is to divide the task of running the overall simulation into five logical components (sub-modules) and then to control the use of these modules. During this process some modules may be used more than once, as indicated by the looping in module 3.0. In general each of the subroutines called will, in fact, be the primary routine for a similar hierarchy. This process continues with finer and finer definition of functions until all computation needs have been satisfied.

B. INPUT(1.0) - Input Data

This module (Figure A.I-1b) controls all data input. There are three main categories of input data:

1. IRNCTL(1.1) - Run Control Input Data - Interactive data are read in module INTAC, and includes such items as the number of weeks to be simulated, the date of the run, the type of run (restart or original) and any run time comments. If a restart is specified, module IRLGIN will read the required restart data from the appropriate files.

2. INETWK(1.2) - Global Parameters - Defined as parameters that are not specified to any one site, global parameters include system parameters (Module ISYSPR) such as the number of sites and service types on the network; and network parameters (Module INETPR) such as network communications costs. This input section is by-passed in case of a restart.

3. ISIDAT(1.3) - Site Specific Data - The data for each site are read in the following sequence:

   a. General Site Data - Includes data such as overall policy information, report selection indicators, number of scheduled up hours, and reliability estimates.

   b. Site Supply Data - Initial descriptions of budgets, system hardware/software, services offered, prices, user support, and capacity impact factors.

   c. Site Demand Data - Includes data such as initial base levels of demand for each user category, user category sensitivity and seasonality factors, and mapping factors for determining service demand by user category.

C. ZSETUP(2.0) - Run Initialization

This module (Figure A.I-1c) performs all pre-run calculations, policy initialization, and initial report generation. The three main sections of ZSETUP are as follows:

1. ZPRERN(2.1) - Pre-run Computations - These include conversion of raw data into appropriate forms, determination of smoothing constants and initial smoothed values, and a variety of preliminary analyses. For "restart" runs, this module is bypassed. The major functions of ZPRERN are:

   a. Initialization of the demand matrix, DR, and the price matrix, PRICE, for every site and service type. Other calculations in ZCOMP include the scaling of input hourly system resource capacities to weekly figures and the computation of initial resource utilizations at every site.

   b. The computation of initial turnarounds and response times for every site and service type in Module ZTURN.

   c. The calculation of network average turnaround, price, and support is performed in special analysis routines ANTCAL and ANEXVL, which are called from ZPRERN.

2. ZIPOL(2.2) - Initialize Standard Policy Vector - See Appendix II-B.

3. ZOUT(2.3) - Pre-run Output - Controls the output of initial information reports for the network as a whole (Module ZNOUT) and each individual site (Module ZSOUT). Initial reports are used for verification of pre-run conditions and as a response point for future comparisons.

D. PROCES(3.0) - Process and Report

This is the controlling module for the period by period (weekly) processing and reporting. Each time period (Figure A.I-1d) it successively calls routines for computation of exogenous changes, supply, demand, load balancing (market analysis), period analyses, and period reports. These routines form the major part of the simulation model and are detailed in the following six sections.

E. XOGEN(3.1) - Exogenous Changes

Exogenous changes are defined as those changes that cannot normally be accomplished by the analytical routines in later modules. On the network level they might include such perturbations as new sites, entire sites going off the network for a period of time, changes in network communication costs, or changes in organizational structure. At the site level, policy changes, configuration changes, or revised demand profiles all fit this category.

Only the overall module structure shown in Figure A.I-1e has been implemented during Phase I of the project. Major use of the module has been for experiments that require unusual perturbations during the run. In general, special routines must be written for this purpose each time this is done. During the later gaming phases of the project, XOGEN will become an interactive routine for on-line input of decisions and policy changes.

F. SUPLY(3.2) - Supply

The output of module SUPLY consists of descriptions of those aspects of a site's offerings that are visible to potential users of the site's computation facilities. These include available services, prices, and levels of support. Such offerings are determined in this module within the guidelines and constraints of supply policies, budgets, and available system hardware and software. Module SUPLY begins with an interpretation of overall site supply policies (3.21) and an evaluation of available budgets and

budget constraints (3.22). The other determinations are then completed in the sequence indicated by Figure A.I-1f. The following sections detail this process.

1. SPOLC(3.21) - Interpret Overall Site Policies - This section of the model interprets the supply policy time flag and translates the current overall policy vector into specific supply areas. The supply policy time flag (Appendix II-G) indicates the status of the current overall policy vector. A site may maintain the policies that it has been using, try new ones for a specified period of time, or permanently change its "standard" policies. The first step in the interpretation of site supply policies is therefore the implementation of any changes in policy specification necessitated by the supply policy time flag. As an example, suppose that XYZ has been following policies that give it a "cost conscious" profile. By using the "time flag" vector, it can specify that it wishes to try out a "marketing" oriented profile for thirteen weeks (for example, during the slow summer quarter). After this time period, XYZ's standard cost conscious policies will be reinstated. If later the site decides that the marketing oriented policies are preferred, these can become XYZ's standard policies.

The second major step is to translate the overall supply policy into specific policy sets. This includes policy sets for budget (ISPOL1), hardware/software (ISPOL2), services available (ISPOL3), pricing (ISPOL4), and support (ISPOL5). As with most decision points, a site can use general policy sets which have been incorporated into the model, or it can access its own routines. Continuing the example, suppose that XYZ is currently following a marketing oriented overall profile. Its policy sets for budget, hardware/software, services available, pricing, and support will all be compatible with this profile.

2. SBUDG(3.22) - Budget - This, and all remaining supply modules, operate in two basic steps -- evaluation of the appropriate policy set, and implementation of that set. The budget

policy (ISPOL1) specified in module SPOLC is evaluated in SBGPOL in order to select the actual budget algorithms and associated parameters to be used in the implementation of this policy. If the budget policy had been specified in the Exogenous Changes module (3.1), SBGPOL would be bypassed.

The second step in the process is the actual computation of the budget allowances according to the specified budget policy in SBGIMP. Suppose, for example, that vector ISPOL1 implied that total budget may not be changed, but individual items could be reallocated. This module might then determine that the budgeted allowance for one of the user categories was exceeded by actual expenditures, but that other categories were well below budget. Following the selected policy, the budget allowance for that user category would be increased, and other categories would be reduced proportionately.

3. SHDSF(3.23) - Hardware/Software - The hardware/software policy (ISPOL2) is evaluated in SHSPOL to determine the configuration change algorithms and their associated parameters. If a site has specified its hardware/software policy in the Exogenous Changes module, this procedure would be omitted.

System utilization is then evaluated and hardware/software capacity modified according to the appropriate hardware/software policy.

4. SERVL(3.24) - Services Available - The available policy set (ISPOL3) is evaluated in SRVPOL to choose the actual services available policy and the associated parameters. If a site has specified its services available policy in the Exogenous Changes module, this section would be bypassed.

The actual evaluation of the demand for the service types not currently offered by the site takes place in SRVIMP. Factors which are considered are: the amount of actual demand for the service type, the turnarounds, response times, and possibly the

amount of unsatisfied demand for that service type. If the site "decides" to offer a new service type, it usually incurs an initial cost of introduction, for which there must be budgeted funds.

5. SPRICE(3.25) - Pricing - Institutions may handle pricing in different ways. Rather than price by service type, most sites charge for each specific resource. For example, XYZ may choose to change its price for CPU time, thus affecting the prices of all services consuming CPU time in proportion to usage of this resource. Assume, however, that a site decides to price by service type. It may pick a policy which raises prices for services that need to be discouraged (according to the same criteria) and lowers them for new offerings or services where usage is to be encouraged.

Unlike the previous service independent supply modules, separate price calculations must be made for each service type. Module SPRPOL determines the pricing policy and associated parameters based on ISPOL4. SPRIMP implements that policy and calculates the price of each service type.

6. SUPOR(3.26) - Support - This section of the model evaluates the support policy set (ISPOL5) for each service type and then computes the actual levels of support that would exist under the selected policies.

G. DMAND(3.3) - Demand

The DEMAND section (Figure A.I-1g) of the Network Simulation model controls the calculation of demand for computer services and its allocation among available suppliers (including the originating site).

1. DPOLC(3.31) - Interpret Overall Site Policy - This module translates the overall site policies (IOPOLC) into specific policies for each user category for demand allocation (IAPOL), truncation, (ITRPOL - imposing budget limits), and user category demand

allocation (IUCPOL).

2. DETER(3.32) - Determine Demand Level - This module determines the desired demand by user category by first computing an expected base level of demand and then modifying this level by seasonality factors and budget limitations.

a. DBASE(3.321) - Determine Base Level - The base level of demand (Figure A.I-1h) for each user category is estimated first as a function of the initial level of demand and an assumed growth profile. This initial estimate is then adjusted to accommodate the realities of present turnaround, price, and support, and the effect of these items on demand. The base level of demand for each user category is a dimensionless number representing the amount (relative to levels at time zero) of overall demand. It will later be translated into service specific demand units. The process of determining the base level each period involves the following equation:

$$BLD = f(I_o, G, T, P, S)$$

where: I_o = initial base level of demand (at time = 0)
G = growth factor for base level
T = turnaround (previous time period)
P = price (previous time period)
S = support (previous time period)

The effect of the growth factor is computed as a function of time, i.e., the week number. The effect of turnaround, price, and support are determined by the relative sensitivity of the user category to each of these factors and the historically expected values of each.

i. DGROW(3.3211) - Demand Base Growth - This function estimates the growth in demand based on some predetermined algorithm. Typical algorithms might

include exponential, linear, the results of a regression analyses, tabular, etc. Presently it is assumed that all demand grows at a compound annual rate of R, where R is input for each user category at each site.

ii. DTURN(3.3212) - Effect of Turnaround on Demand - This function calculates an adjustment to the base level of demand based on turnaround. The actual turnaround is compared to the historically achieved turnaround. If actual is worse than expected, then this causes a reduction in demand. If the actual turnaround is better than the expected turnaround, demand will be increased. The absolute magnitude of this change in demand level depends on the difference between actual and expected and the sensitivity of the user category to this factor.

iii. DPRICE(3.3213) - Effect of Price on Demand - This function calculates an adjustment to the base level of demand based on price. As in turnaround, the actual price is compared to the expected value and the increase or reduction is determined by the difference and the user's sensitivity to price.

iv. DSUPP(3.3214) - Effect of Support on Demand - This function calculates the adjustment to demand based on support. Again, actual support is compared to expected levels for determining the difference and the effect.

b. DSEAS(3.322) - Seasonalize Demand - This module performs the seasonalization of the base levels of demand for each user category. This is accomplished by multiplying the computed base level of demand by a monthly seasonality factor (assuming a 13 month year, 4 weeks

per month), i.e., by the expected percentage variation. The seasonality factors are stored in tabular form for each user category at each site.

c. DTRUNC(3.323) - Truncate Demand due to Budget Limitations - The following general procedure is used to ensure that calculated demand is compatible with the available budget:

i. Spread (map) base level of demand by user category into service type demands. A system utility routine performs this computation (USTMAP). Basically, it is assumed that for each user category, proportional usage of each service type remains constant. Hence the spread is a simple proportionate mapping.

ii. Find the approximate cost to run these jobs using current prices.

iii. Add costs over all service types to get the expected total expenditure for the user category.

iv. Use a budget truncation policy (Appendix II.E-3) to determine if budget constraints will be violated and, if so, the demand estimate should be reduced (truncated).

3. DALLO(3.33) - Allocate Demand Among Candidate Sites - As with all policy areas, this is a two stage procedure. The allocation policies are first evaluated in module 3.331. Based on these policies, the available sites are examined and the demand is allocated (module 3.332). This module is exercised for each user category.

a. DAPOL(3.331) - Evaluate Allocation Policies - The first step (Figure A.I-1j) is to determine for the given user category the restrictions on where these

users are allowed to go to satisfy their demand. The policies at the user category level are evaluated in module DUPOL.

Before demand can be discussed at the service type level, it is necessary to map the user category base level of demand (determined previously in module 3.32) into service type demands. Module DMAP accomplishes this via the system utility routine USTMAP. Finally, module DSPOL permits the definition of service-specific allocation policies if these are desired. Most sites will use the same policy for all work attributed to a given user.

b. DSALLO(3.332) - Select and Allocate by Service Type - Once all allocation policies have been defined, the actual site selections and allocations must be accomplished (Figure A.I-1k). This involves simultaneous consideration of the site selection methods and the user category allocation restrictions, and is controlled by module 3.3321.

i. DRATE(3.3321) - Compute Allocations of Demand - The rating and selection model allocates service specific demands to the best available site for each user category. This is done in three steps:

DCOSET(3.33211) - Set Rating Coefficients - The coefficients used for the rating of available sites (including in-house) are determined by this module. There are coefficients for price, turnaround, support, and momentum, i.e., past demand.

DUCRES(3.33212) - Impose User Restrictions - The restrictions on available sites for demand allocation at the service type level are imposed in this module.

DPOL1(3.33213) - Rate Sites and Allocate Demand - The policies set in module DSPOL (3.3313), in combination with the coefficients from DCOSET, and user category restrictions from DUCRES, are used in this module. Every available site is rated, and the demand is allocated on the service type level to the best sites. Limits as to the number of allocatible sites are set as a function of policy.

ii. DSMTH(3.3322) - The calculation of the expected values of turnaround, price, and support for the user category. This is accomplished by first computing the actual values for each service type, summing these results to obtain a total figure, and then dividing this total by the number of services demanded by the user category.

iii. DSDR(3.3323) - Update Demand Matrix (DR) - The allocations determined in module 3.33213 are added to the appropriate elements of the Demand Requested (DR) matrix. Values of this matrix represent a running sum of all demand. This matrix is complete only after the entire demand section of the model (3.3) has been completed.

H. MARKET (3.4) - Market Analysis

This routine (Figure A.I-11) controls the allocation of each site's system resources among the requested demands from all network and internal users. It takes into account supply constraints, scheduling priorities, communication loads, etc. There are three major segments:

1. MALLO(3.41) - Supply Allocation - The allocation of system resources at every site is performed (Figure A.I-1m) as follows:

a. MSUM(3.411) - Demand Summation - All service demands originating at that site and all service specific de-

mands directed to the site are calculated in this routine. These demands are then mapped into system resource requests using the special routine RIPMAP (service type to resource mapping).

b. MPOLEV(3.412) - Policy Evaluation - This routine assigns the appropriate segment of the site's overall policy to its market policy, ISPOL. This is actually done in MOPOLC. Upon determination of this policy, MAPOL (Allocation of Supply Policy) determines the appropriate market cutback algorithm and parameters.

The routine MCUT (3.41221) is used to compute any over-utilizations of system reources and to set the percentages to cut service specific demands.

c. MREAL(3.413) - Allocation of Supply - This routine controls the actual cutbacks of service specific demand. The method of demand cutting is a function of policy and specific site constraints (performance contracts, etc.). (See Appendix II-F).

2. MLOAD(3.42) - Estimate Turnaround - This routine will estimate service type turnarounds at every site for the purpose of allocating "network" demand. This module is a stub at the present time, since the network site has not yet been implemented.

3. MNET(3.43) - Allocate Network Demand - This routine will allocate network demand to the best available sites. This module is a stub at the present time, since the network site has not been implented.

I. ANALY(3.5) - Period Analyses

This routine (Figure A.I-1n) controls the period summary analyses for the individual sites and for the network. It is composed of two major segments. The calculations done here are available for

reporting purposes in the report segment (4.0).

1. ASITE(3.51) - Site Analyses - The controlling routine for the period analyses performed by every site calls the following modules:

a. ASUM(3.511) - General Computations - Computer site averages, summations, and other summary data. For example, it calculates the site average price over all service types for each site.

b. ATURN1(3.513) - Turnaround Calculation - Turnarounds for every site and service type are determined in this module. Every site has its own unique turnaround calculations sequenced as modules ATNS01 through ATNS20 which are called by ATURN1 as illustrated in Figure A.I-1o.

c. APUS(3.513) - Support Level - This routine computes the per unit dollar level of support that users will see. These support levels msut be determined for every service offered at the site.

d. ASTAFF(3.514) - Site Staffing - Currently a stub, this routine is available for future studies concerning computer center staffing at any site.

e. ACOMM1(3.515) - Site Communications - The total communications load (both from the network and to the network) is calculated in this routine. This is summed over all services. Note that work satisfied in-house will not be included in the communications load since this is not sent out to the network.

f. AINEX(3.516) - Income/Expense - Controls the computations of income and expenses for every site during the past period. Yearly cash flow figures are updated to include the new figures. These calculations include internal income, external income, other income, communication charges, supply expenses, and total user expenditures.

2. ANETWK(3.52) - Network Analyses - Statistical analyses on the network level are controlled by this routine. The order of flow is:

a. ANTCAL(3.521) - Network Averages - Network statistics, including average service specific turnaround and standard deviations about the average, smoothed turnarounds, and average network prices, are calculated in this module. These figures are based only on network sites offering the particular service type in question.

b. ANEXVL(3.522) - Network Expected Values - Smoothed (expected) values for turnaround, price, and support are computed for every service type in this routine. Each smoothed value is a function of the current week's statistics and the previous smoothed values.

c. ACOMM2(3.523) - Network Communications - This routine is for analyses of network communications loads. It is a stub at present.

J. REPORT(3.6) - Report

This routine generates all period reports (Figure A.I-14). Every site has the option of specifying any of the following types of period reports and the intervals at which they should be generated:

1. RSITE(3.61) - Site Reports - Site specific reports are

generated under the control of this module. These reports include financial reports such as budget, cash flow, and income/expenditure reports; special reports such as site turnaround, site utilization, and site policy reports; and service specific reports such as turnaround and price by service type at every site on the network.

2. RNET(3.62) - Network Reports - Reports on network flows are generated by this routine. These reports include communications, cash flows, and other special reports.

K. COMPUT(4.0) - Summary Computations

After all processing is complete for each period, this routine performs various analyses concerning the entire length of the simulation. Time series analyses and tabulation of network configuration changes are examples of the types of computations that may be done. Both this module and the following one are conceptual representations in the module flow, since these calculations and reports will most likely be done off-line after the simulation run has completed.

L. GENREP(5.0) - Generate Summary Reports

This routine controls the generation of summary reports for the entire simulation run. These reports fall into two categories:

1. Summary reports on individual site behavior including reports on communications, service, capacity utilization, and summaries of the reports produced by RSITE.

2. Summary reports on network behavior patterns including summaries of the reports produced by RNET. As mentioned above, these reports will probably not be produced on-line but will be generated from the LOG file.

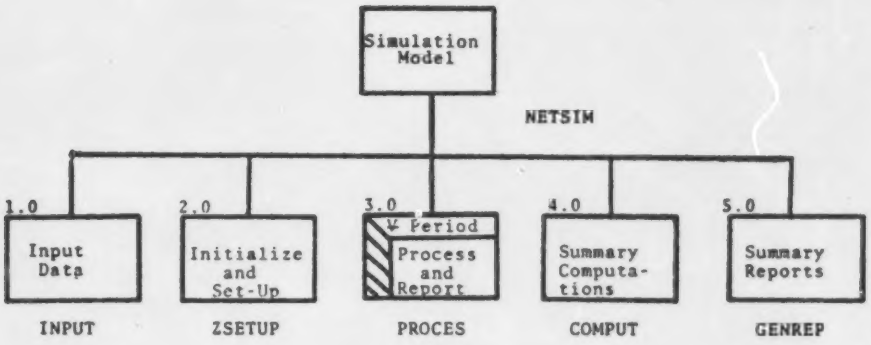

Simulation Model
NETSIM
1.0
Input Data
INPUT
2.0
Initialize and Set-Up
ZSETUP
3.0
Period
Process and Report
PROCES
4.0
Summary Computa-tions
COMPUT
5.0
Summary Reports
GENREP

Figure A.I-1a

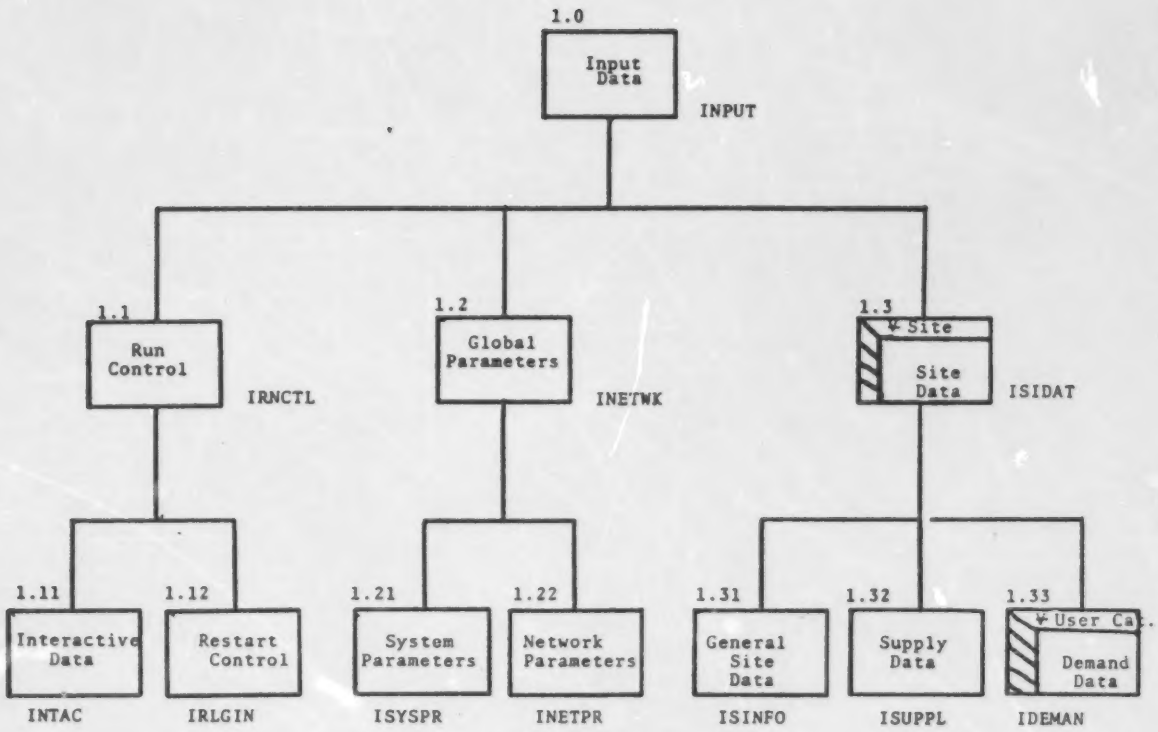

1.0
Input Data
INPUT
1.1
Run Control
IRNCTL
1.2
Global Parameters
INETWK
1.3
Site
Site Data
ISIDAT
1.11
Interactive Data
INTAC
1.12
Restart Control
IRLGIN
1.21
System Parameters
ISYSPR
1.22
Network Parameters
INETPR
1.31
General Site Data
ISINFO
1.32
Supply Data
ISUPPL
1.33
User Ca
Demand Data
IDEMAN

Figure A.I-1b

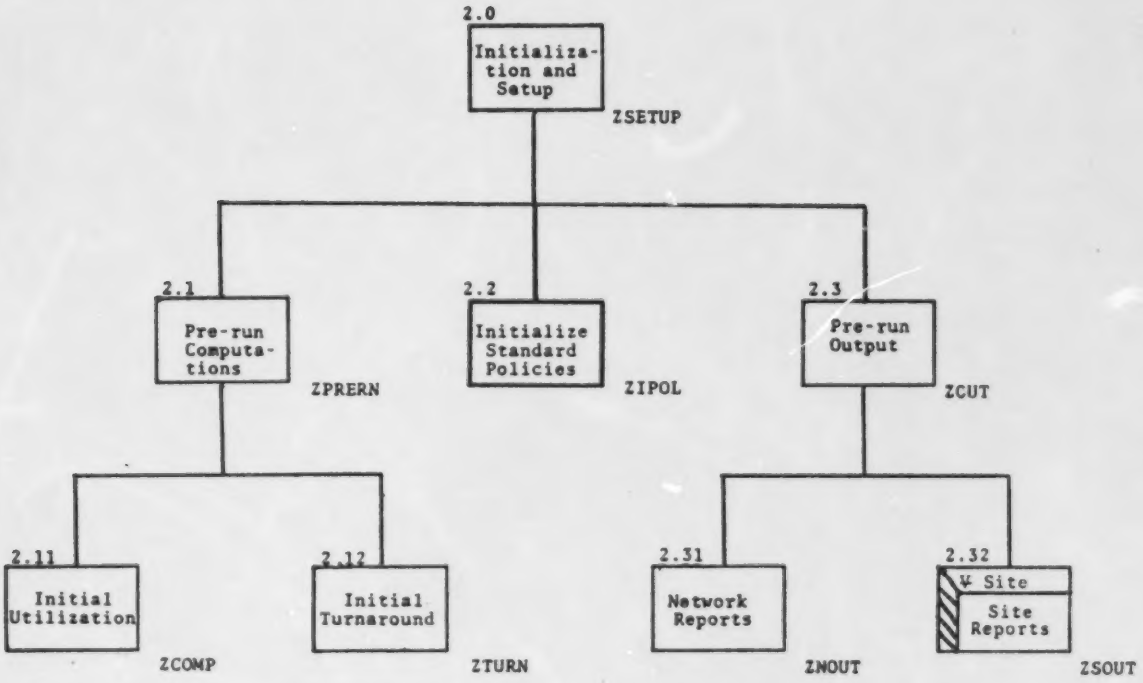

2.0
Initializa-
tion and
Setup
ZSETUP
2.1
Pre-run
Computa-
tions
ZPRERN
2.2
Initialize
Standard
Policies
ZIPOL
2.3
Pre-run
Output
ZOUT
2.11
Initial
Utilization
ZCOMP
2.12
Initial
Turnaround
ZTURN
2.31
Network
Reports
ZNOUT
2.32
Ψ Site
Site
Reports
ZSOUT

Figure A.I-1c

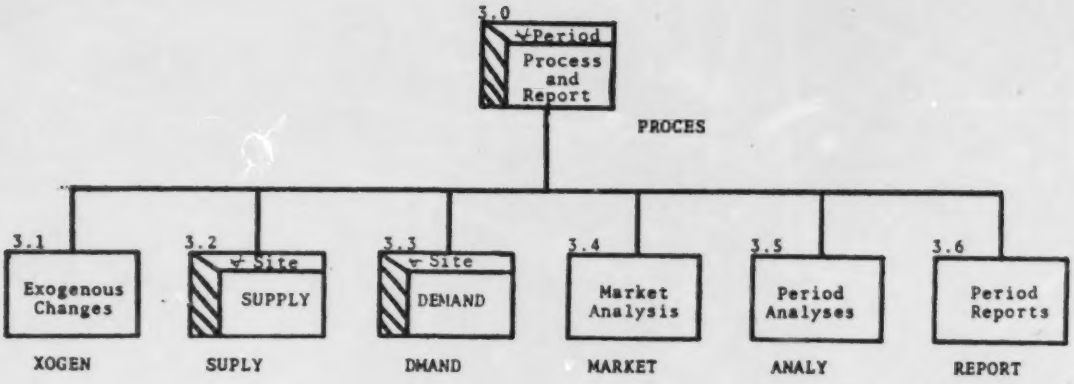

3.0
∀Period
Process and Report
PROCES
3.1
Exogenous Changes
XOGEN
3.2
∀ Site
SUPPLY
SUPLY
3.3
∀ Site
DEMAND
DMAND
3.4
Market Analysis
MARKET
3.5
Period Analyses
ANALY
3.6
Period Reports
REPORT

Figure A.I-1d

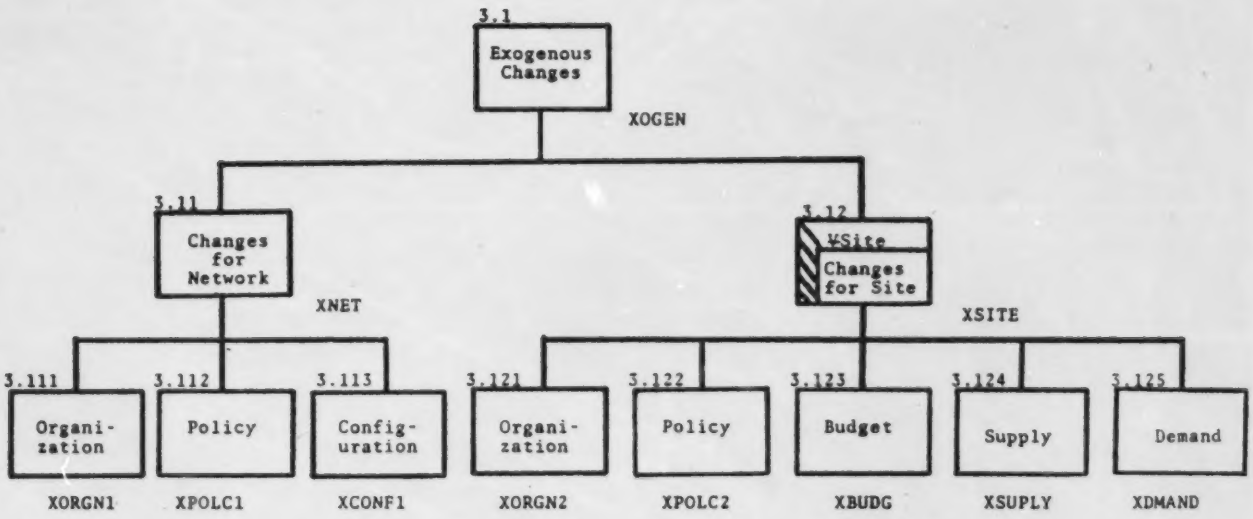

3.1
Exogenous Changes
XOGEN
3.11
Changes for Network
XNET
3.12
¥Site
Changes for Site
XSITE
3.111
Organi-zation
XORGN1
3.112
Policy
XPOLC1
3.113
Config-uration
XCONF1
3.121
Organi-zation
XORGN2
3.122
Policy
XPOLC2
3.123
Budget
XBUDG
3.124
Supply
XSUPLY
3.125
Demand
XDMAND

Figure A.I-1e

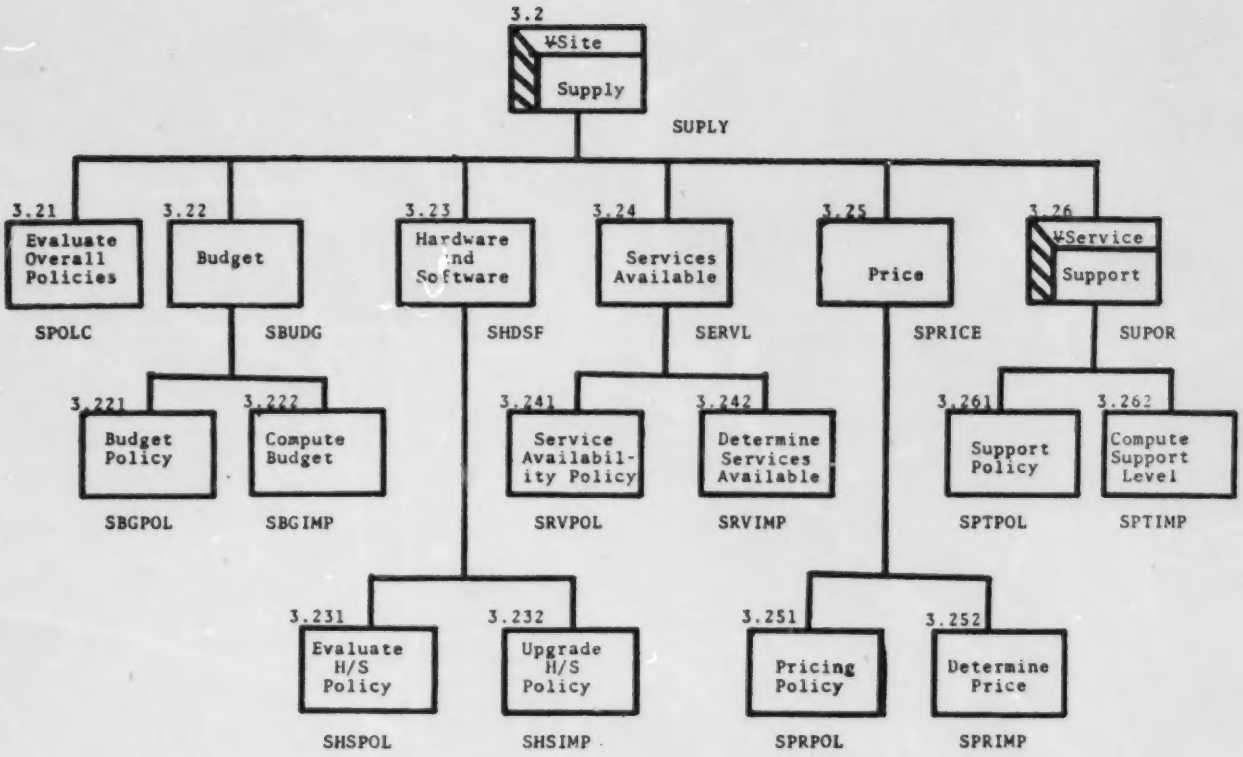

3.2
¥Site
Supply
SUPLY
3.21
Evaluate Overall Policies
SPOLC
3.22
Budget
SBUDG
3.23
Hardware and Software
SHDSF
3.24
Services Available
SERVL
3.25
Price
SPRICE
3.26
¥Service
Support
SUPOR
3.221
Budget Policy
SBGPOL
3.222
Compute Budget
SBGIMP
3.241
Service Availability Policy
SRVPOL
3.242
Determine Services Available
SRVIMP
3.261
Support Policy
SPTPOL
3.262
Compute Support Level
SPTIMP
3.231
Evaluate H/S Policy
SHSPOL
3.232
Upgrade H/S Policy
SHSIMP
3.251
Pricing Policy
SPRPOL
3.252
Determine Price
SPRIMP

Figure A.I-1f

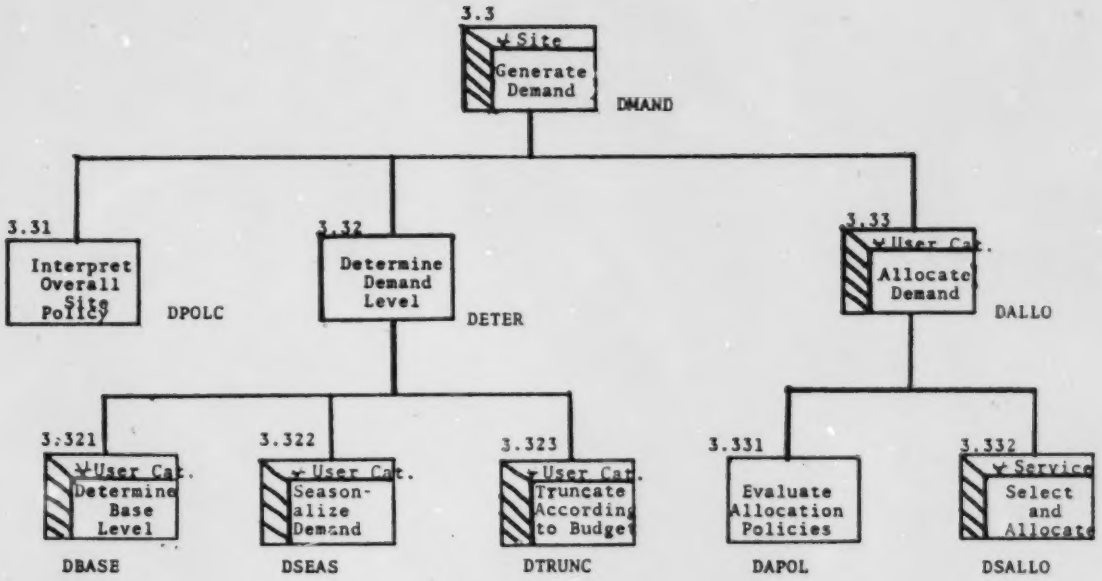

3.3
Site
Generate Demand
DMAND
3.31
Interpret Overall Site Policy
DPOLC
3.32
Determine Demand Level
DETER
3.33
User Cat.
Allocate Demand
DALLO
3.321
User Cat.
Determine Base Level
DBASE
3.322
User Cat.
Season-alize Demand
DSEAS
3.323
User Cat.
Truncate According to Budget
DTRUNC
3.331
Evaluate Allocation Policies
DAPOL
3.332
Service
Select and Allocate
DSALLO

Figure A.I-1g

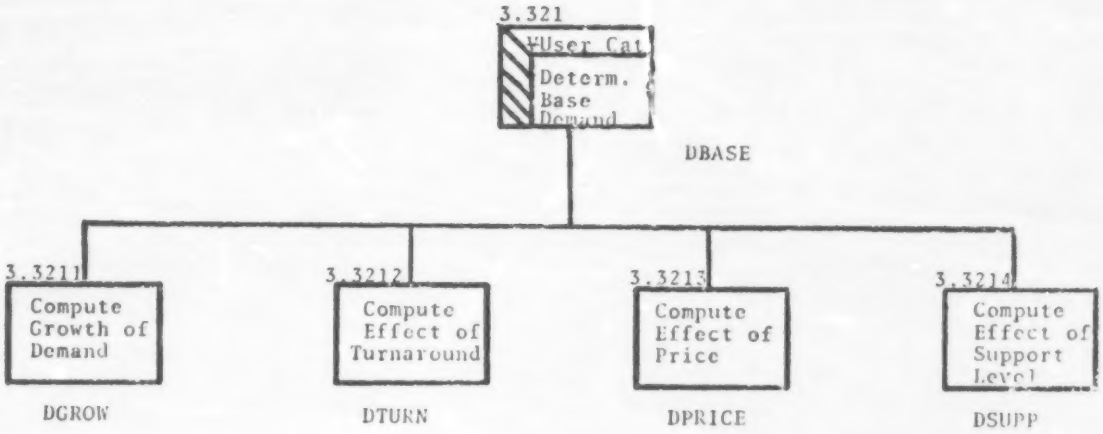

3.321
#User Cat.
Determ.
Base
Demand
DBASE
3.3211
Compute
Growth of
Demand
DGROW
3.3212
Compute
Effect of
Turnaround
DTURN
3.3213
Compute
Effect of
Price
DPRICE
3.3214
Compute
Effect of
Support
Level
DSUPP

Figure A.1-1h

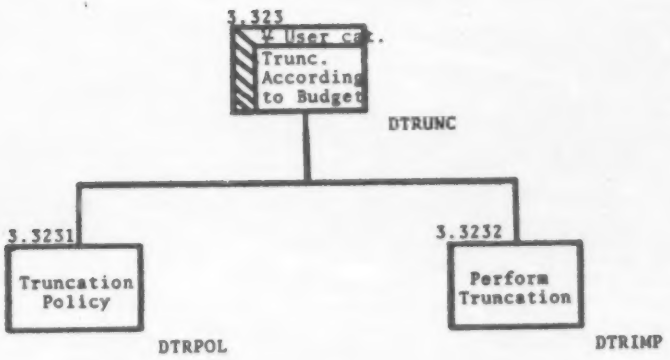

3.323
Trunc.
According
to Budget
DTRUNC
3.3231
Truncation
Policy
DTRPOL
3.3232
Perform
Truncation
DTRIMP

Figure A.I-11

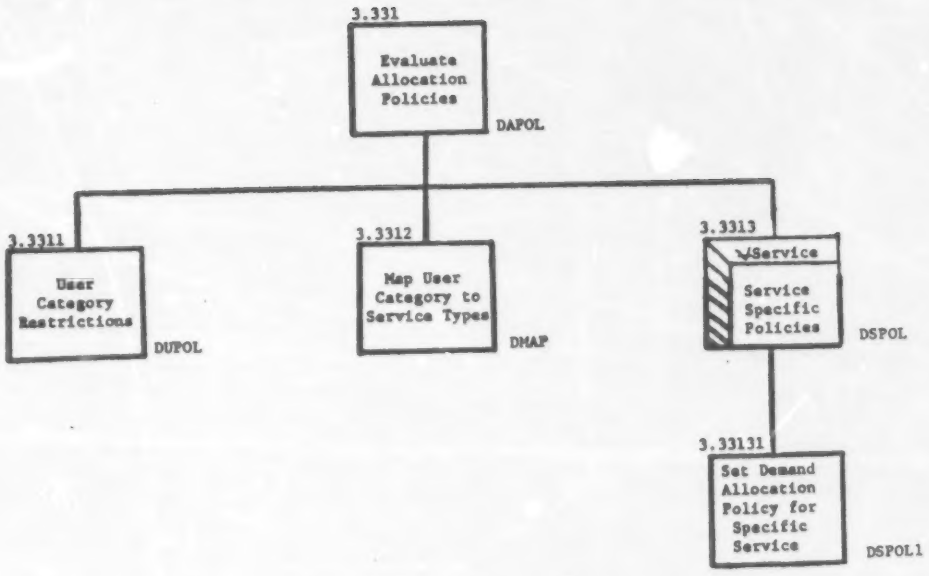

3.331
Evaluate Allocation Policies
DAPOL
3.3311
User Category Restrictions
DUPOL
3.3312
Map User Category to Service Types
DMAP
3.3313
√Service
Service Specific Policies
DSPOL
3.33131
Set Demand Allocation Policy for Specific Service
DSPOL1

Figure A.I-1j

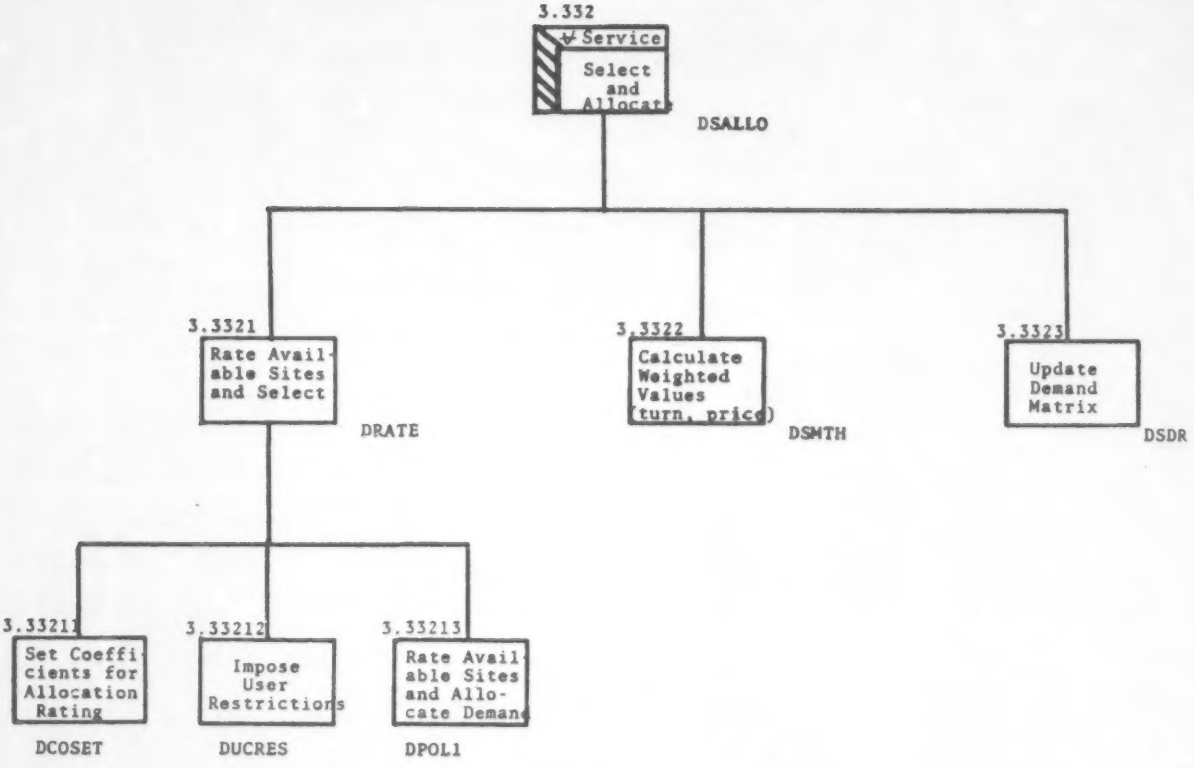

3.332
Service
Select and Allocat
DSALLO
3.3321
Rate Available Sites and Select
DRATE
3.3322
Calculate Weighted Values (turn, price)
DSMTH
3.3323
Update Demand Matrix
DSDR
3.33211
Set Coefficients for Allocation Rating
DCOSET
3.33212
Impose User Restrictions
DUCRES
3.33213
Rate Available Sites and Allocate Demand
DPOL1

Figure A.I-1k

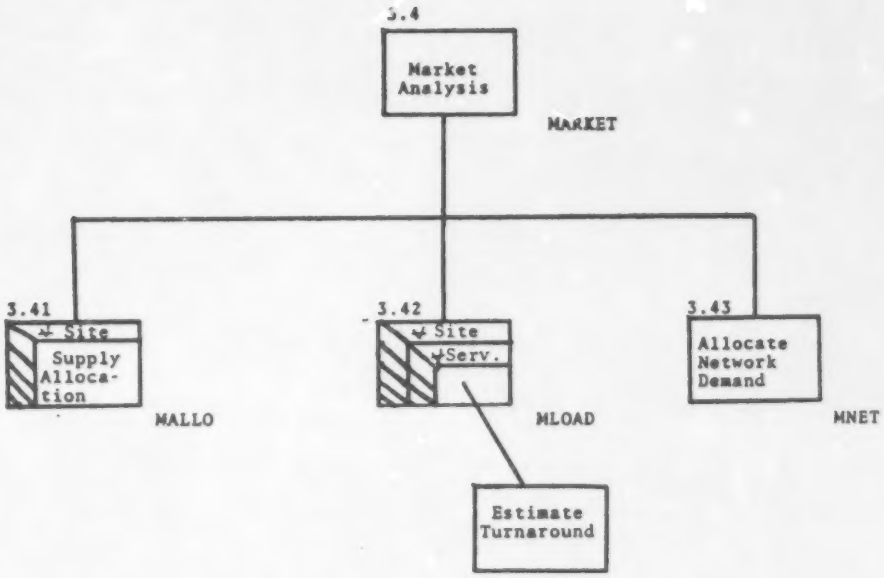

3.4
Market Analysis
MARKET
3.41
+Site
Supply Alloca-tion
MALLO
3.42
+Site
+Serv.
MLOAD
3.43
Allocate Network Demand
MNET
Estimate Turnaround

Figure A.I-11

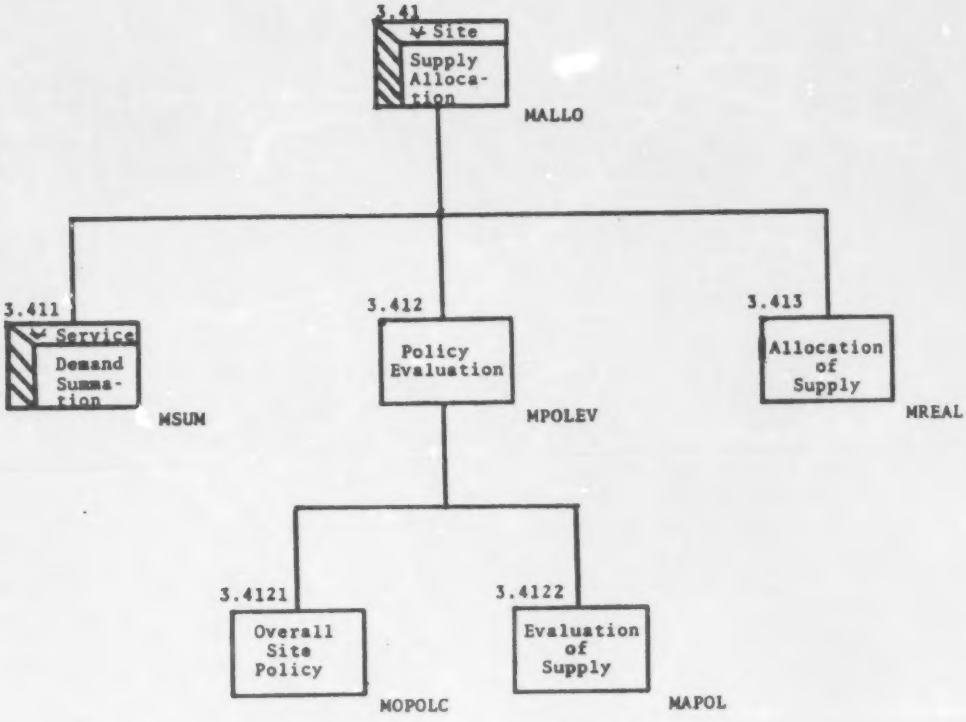

3.41
Site
Supply Alloca-tion
MALLO
3.411
Service
Demand Summa-tion
MSUM
3.412
Policy Evaluation
MPOLEV
3.413
Allocation of Supply
MREAL
3.4121
Overall Site Policy
MOPOLC
3.4122
Evaluation of Supply
MAPOL

Figure A.I-1m

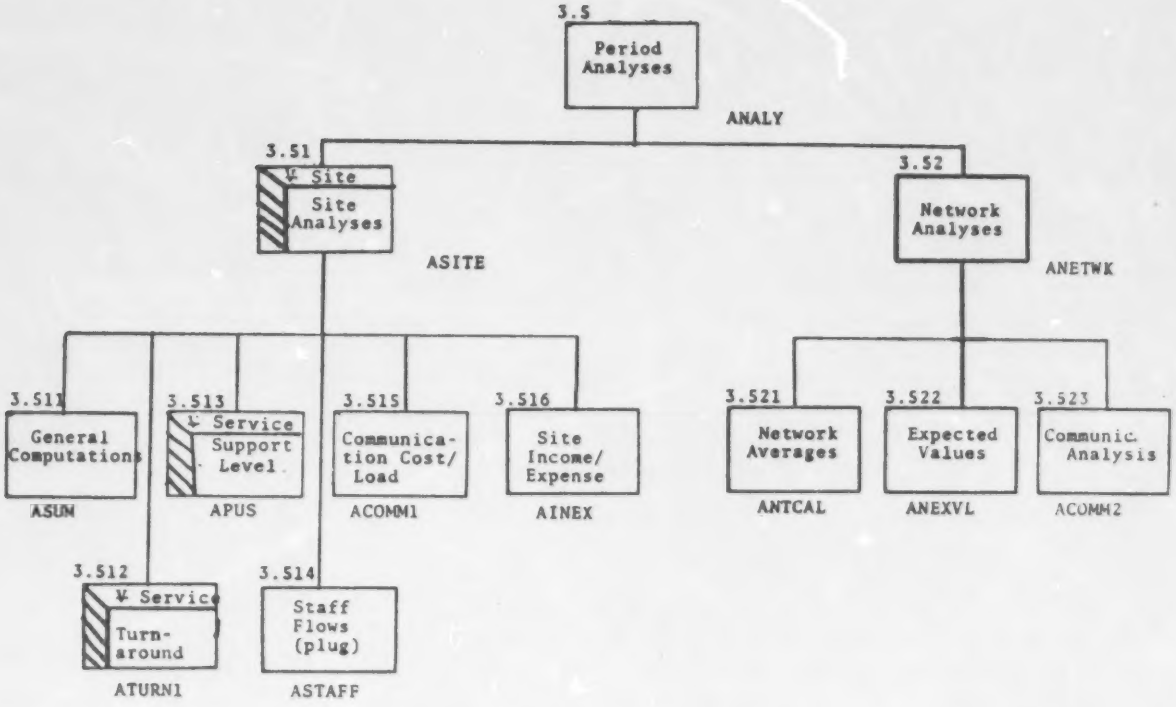

3.5
Period Analyses
ANALY
3.51
Site
Site Analyses
ASITE
3.52
Network Analyses
ANETWK
3.511
General Computations
ASUM
3.513
Service
Support Level
APUS
3.515
Communication Cost/ Load
ACOMM1
3.516
Site Income/ Expense
AINEX
3.521
Network Averages
ANTCAL
3.522
Expected Values
ANEXVL
3.523
Communic. Analysis
ACOMM2
3.512
Service
Turn-around
ATURN1
3.514
Staff Flows (plug)
ASTAFF

Figure A.I-1n

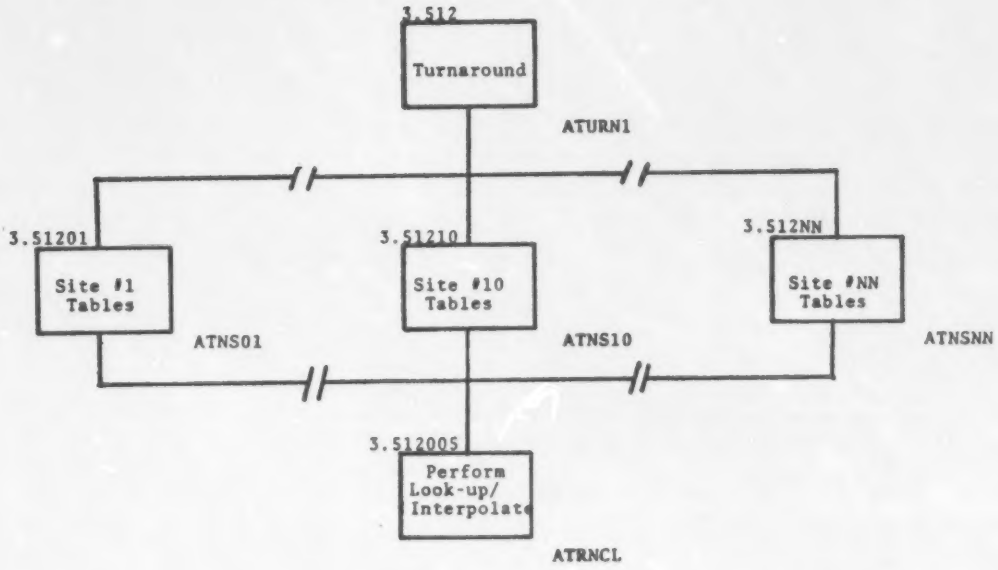

3.512
Turnaround
ATURN1
3.51201
Site #1
Tables
3.51210
Site #10
Tables
3.512NN
Site #NN
Tables
ATNS01
ATNS10
ATNSNN
3.512005
Perform
Look-up/
Interpolate
ATRNCL

Figure A.I-1o

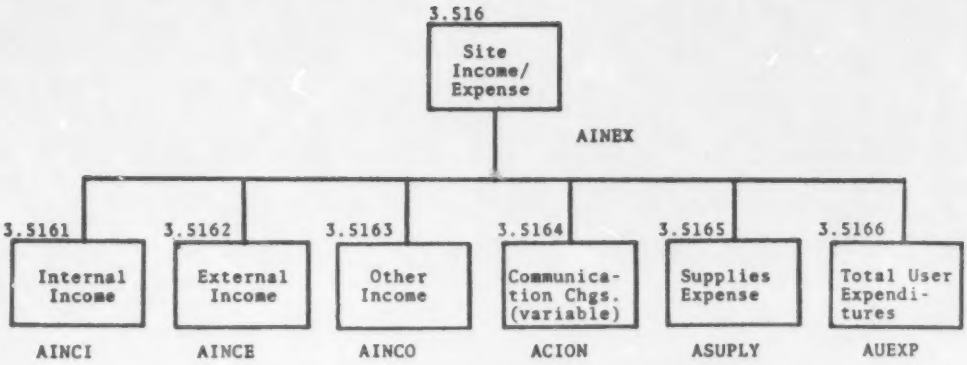

3.516
Site Income/ Expense
AINEX
3.5161
Internal Income
AINCI
3.5162
External Income
AINCE
3.5163
Other Income
AINCO
3.5164
Communica-tion Chgs. (variable)
ACION
3.5165
Supplies Expense
ASUPLY
3.5166
Total User Expendi-tures
AUEXP

Figure A.I-1p

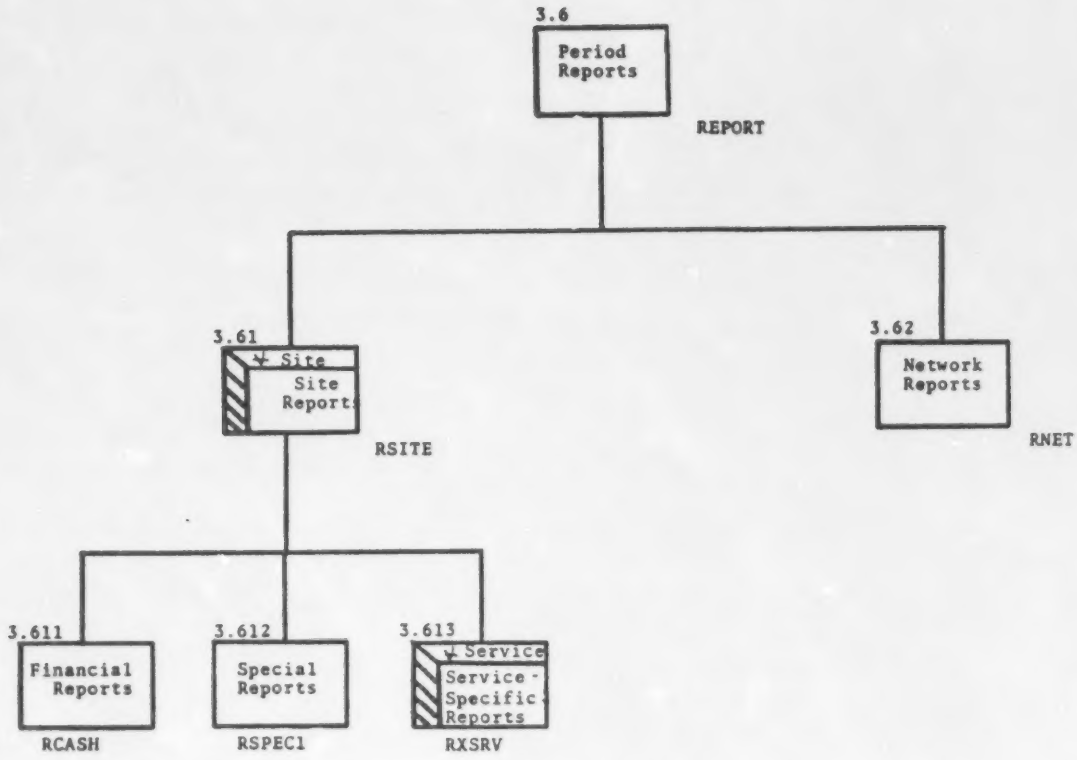

3.6
Period Reports
REPORT
3.61
Site
Site Reports
RSITE
3.62
Network Reports
RNET
3.611
Financial Reports
RCASH
3.612
Special Reports
RSPEC1
3.613
Service
Service-Specific Reports
RXSRV

Figure A.I-1q

Appendix II

Model Policies and Representations

A. Policies and Practices - Overview

The model was designed to be highly parameterized and "policy-driven." Any discussion of the model must therefore emphasize descriptions of the way policies are incorporated into the model and used within the model. Every simulation run begins with the input of appropriate policies, practices, and/or management decisions for each site. These policy selections control various decision points within the model so that the actions taken will accurately represent that site (or a viewpoint which that site might wish to test).

The policy selections determine many aspects of site and network activity:

- how a site handles its budget
- when and to what extent it makes changes in its configuration
- which service types it is going to offer for any period
- the prices to be charged for any resource and service type
- the level of user support it will provide for a given resource type
- the level of demand generated by local users
- budget constraints on the demand which it generates
- allocations of available resources when the site is over-loaded (which sites get priority because of previous dealings or commitments, etc.)

In general, policies are conceptually dealt with from two levels: an overall site profile and supply, demand, and market (load leveling). The "overall profile" is reflected by a consistant selection of supply, demand and market policies to represent an institution's posture relative to a network. A single vector (IOPOLC) is used to carry indications of the major policies in each area. Several vectors are used to house the second level

policies - supply, demand and market. Detailed descriptions of how each of these perspectives is implemented are given in Sections B through F of this Appendix.

The discussions in this Appendix focus on the structure of policy representations and the options available within the model. More specific information relative to programming conventions and procedures for modifying the model appears as Appendices III through V.

B. Overall Site Profiles

The overall profile of any site is reflected by the policies selected in the areas of supply determination, demand generation and allocation, and load leveling (market). Sites can be represented as being predominantly cost conscious, profit oriented, marketing oriented (many services offered with good support), user sensitive, etc. The model provides for alterations of overall site profiles (policies) during the course of the simulation. There are two policy vectors used to store these overall policy representations:

IOPOLC - the current overall trial policy for a site
ICOPOL - the standard overall policy for a site

Each of these vectors contains the full set of top-level policy numbers, associated parameters, and time flags (Appendix II-G).

1. Current Policy Vector (IOPOLC) - For each of the major policy areas, a list of available decision rules is maintained within the model. The current overall policy vector, IOPOLC, contains either the numbers of appropriate policies from these lists, or indications to use site specific routines or procedures. This vector is specified for every site by the INPUT section (module 2.0) of the model. At the present time, the initial policies remain in effect for the entire simulation run. In later project phases, current policies will be changeable on either a temporary or perma-

nent basis in the "Exogenous Changes" module (3.1). The current overall site policy vector is of the form:

IOPOLC(ISITE,I)

where: ISITE = site number
I = 1 to 15

Contents of the vector for permissible values of I are:

I	Description
1	Supply policy affecting budget (code number - see Appendix II-G).
2	Supply policy affecting hardware/software.
3	Supply policy affecting services available.
4	Supply policy affecting pricing.
5	Supply policy affecting support and user services.
6	Demand policy affecting cuts in demand at the user category level due to budget restrictions.
7	Demand policy affecting user restrictions on demand allocation.
8	Demand policy affecting service specific demand allocation.
9	Market policy (load leveling).
10	Variable 1 - Available for use by any policy. Currently used to indicate the number of outside sites to which a given site may allocate its demand.
11	Variable 2 - Available for use by any policy. Currently used to indicate the maximum deficit permitted by a site.
12	Variable 3 - Available for use by any policy. Currently not used.
13	Time flag for the supply policies (see Appendix II-G).
14	Time flag for the demand policies.
15	Time flag for market policies.

2. Standard Policy Vector (ICOPOL) - The standard overall policy (ICOPOL) is the general profile describing each site's "normal" behavior. It is initialized for the current policy to the INPUT data, and remains constant throughout the simulation.

In Project Phase III, sites will have the option of specifying new policies in the "Exogenous Changes" module. Maintaining the vector ICOPOL, in effect, permits a site to try temporary policies during the simulation run using the IOPOLC vector as described in the previous section. The periods of time during which the temporary policies remain in effect are specified with time flags (described in Appendix II-G) in the "Exogenous Changes" module. When the specified time period has elapsed, the standard policies are restored. Supply, demand, and market policies each have separate time flags. The standard overall site policy vector is of the form:

ICOPOL(ISITE,I)

where: ISITE = site number
I = 1 to 12

For each site the 12 values of ICOPOL are equivalent to the first 12 values of IOPOLC (see above section).

C. General Supply Policies

Supply policies for determining the budget and hardware/software configuration must be specified for each site. These two supply policy categories relate to the overall institution computer operation and are not specific for individual service types. The vector used is of the form:

IPOLS2(ISITE,J,I)

where: ISITE = site number
J = policy type
1 - budget policy
2 - hardware/software policy

I = policy indicator
1 - policy number
2 - parameter for that policy

1. Budget (J=1) - Sites will have a variety of options for evaluating their budgets (see Section IV-H). The following typical budget policies are currently implemented:

a. Fixed budget - The budget is not changed under any circumstances. This policy might be used by a site whose budget is determined on an annual basis by, say, the state legislature.

b. If actual expenditures exceed a segment of the budget, the budget is raised for that segment and a segment in which the budgeted amount exceeds actual expenditures is reduced. This can represent, for example, a site with a fixed total budget but flexibility to reallocate internal budget lines. Decisions can be based on actual expenditures to date, projected expenditures for the year, etc.

2. Hardware/Software (J=2) - Either actual or projected usage of a site's resources are compared with the capacity of each resource in order to determine if any configuration changes are appropriate. The following hardware/software policies are typical:

a. The system should be highly utilized -- i.e., upgrade the system only when necessary. The system is likely to be down graded when appropriate, e.g., if certain resources are very under-utilized. This policy represents a site with a "cost conscious" profile, and equipment that is primarily on short term lease or rental.

b. Project usage optimistically - system upgraded when appropriate, but rarely downgraded. This policy might be used by a site with a "growth intensive" profile.

c. The system is upgraded with great reluctance, and rarely downgraded. Budgetary constraints are followed. Policy is similar to "a" except that equipment is purchased on a long-term lease.

D. Service-Specific Supply Policies

These policies operate in the same manner as the policies described in the previous section, except in some cases they may not be the same for all service offerings at the site. Policies must therefore be provided at the service type level. Areas of concern include determining which services to offer, prices, and levels of support for each service type. The supply policy vector for service-specific policies is:

IPOLS1(ISITE,KTYPE,J,I)

where: ISITE = site number
KTYPE = service type
J = policy type
1 = services available policy vector
2 = pricing policy vector
3 = level of support policy vector

I = policy indicator
1 - policy set number
2 - parameter for policy set

1. Services Available (J=1) - It is assumed that there is an initial cost for introducing each new service type. Currently it is assumed that this cost is the same for every site. The policy used may vary with the particular service type, -- e.g., file manipulation and reporting must be offered, APL need not be. The following policies for determining services available are among the options that currently may be selected:

a. A new service type is offered only if there is a large unsatisfied demand for that service on the network and there is money available in the appropriate section of the budget. This policy is typically used by a site with a "cost conscious" profile.

b. A new service type is offered if there is a perceived long-term demand for it (i.e., demand is increasing).

Budget constraints are disregarded. This policy would be used by a site with a "growth intensive" profile.

c. A new service type is offered if there is an immediate demand for it. Budgetary constraints are loosely followed, but emphasis is on comparing expected returns with cost. This policy might reflect a site with a "marketing oriented" profile.

2. Pricing (J=2) - A number of different pricing policies are available in the model. Most sites will price by resource, changing the prices for critical resources, under-utilized resources, etc., so as to encourage efficient overall system usage. A change in the resource charge will automatically affect the prices for all service types using that resource. Note that a site following a resource-based pricing strategy will not have service-specific pricing policies. Some pricing policies which use the resource charging alternative are:

a. The price of a resource is raised if it is over-utilized, and lowered if it is under-utilized. Cutoff points for utilizations may be determined by specification of the parameter (I=2).

b. Resource prices are modified with the objective of matching total estimated income with total estimated expenditures.

Other sites may price directly by service offered, ignoring resource charges. This type of policy would allow a site, for example, to fix the average price of a connect hour or a fast student compiler (i.e., WATFIV). Some pricing policies which use either the service-specific pricing alternative or the resource charging alternative are:

a. Prices are raised above the network average if system utilization is high and actual revenue is lower than projected revenue. Prices are lowered towards, but above, the network average if utilization is low. This policy is typically used by a site with a "cost conscious" profile.

b. Prices are raised if utilization is high. Prices are lowered if utilization is low. This policy is used by a site with a "growth intensive" profile.

c. Prices are raised to network average if utilization is high and actual revenue is lower than projected revenue. Prices are lowered towards network average if utilization is low. This policy is used by a site with a "marketing oriented" profile.

3. Level of Support (J=3) - Each site provides some level of support for each service type offered. This represents the auxiliary services which may be available to the user of a site. At the present time, support is represented as a single dollar level for each service type at the site (see Section IV-C.4). The following policies for determination of support level are among those available:

a. Try to stay slightly below the network average while closely following the budget. This policy is typically used by a site with a "cost conscious" profile.

b. Service-type dependent (i.e., good support to some service types, and little or no support to other types). This policy may be used by a site with a "growth intensive" profile to encourage appropriate service type usage.

c. Keep support levels above network averages. Disregard budgetary constraints. This policy might be used by a site with a "marketing oriented" profile.

Note: There are many aspects of support to be considered, e.g., manual preparation, printing, on-line tutorials, CAI, and advisement. These may all be represented by the dollar costs as described in each site's budget. Fixed costs would include advisors' salaries, manual preparation, etc., while variable costs are associated with operations such as printing manuals, phone calls, computer time used for support functions, etc. The current representation of support in the site's budget combines the fixed and variable portions. Although "quality of support" as perceived by the user is obviously heavily influenced by the type of support provided, it is reasonable to assume that, on balance, sites will provide the most appropriate form of support for each service type. User decisions (see Demand Policies - Appendix II-E) can therefore be based on the amount of money spent on the support function.

E. Demand Policies

1. User Category Budget Constraints (IPOLDT) - This module compares user category expenditures with the budgeted amounts and, if necessary, "truncates" the demand estimation so that it is compatible with available funds. The major issue in this segment concerns the definition of "available funds." The policy vector for the determination of a site's budget truncation method is:

IPOLDT(ISITE,IUCAT,I)

where: ISITE = site number
IUCAT = user category
I = 1 or 2
1 - budget truncation policy
2 - truncation policy parameter

Typical policies for determining the cutoff point are:

a. Never allow a weekly expenditure to exceed 1/52 of the annual budget. (This trivial policy is used only for model testing).

b. Never allow the cumulative expenditures at the end of week n to exceed n/52 times the annual budget by more than X%, where X is the parameter associated with this policy.

c. Do not allow cumulative expenditures to exceed n/52 times the annual budget by more than X% of the remaining funds i.e., X% of $\frac{52-n}{n}$ times the annual budget where X is as specified above.

d. Place no restrictions on expenditures.

e. Same as "b" but applied to all user categories combined, i.e., only total expenditures.

f. Same as "c" but applied to all user categories combined

2. User Category Allocation Restrictions (IPOLUA) - After the total demand for each user category is determined, this demand must be allocated to particular supplier sites. Some user categories at the site may not be permitted to use the services offered at certain other sites. For example, a student at ABC University may not be able to send any work outside. A faculty member, on the other hand, may be able to use sites XYZ, AAA, or ZZZ for his work. These restrictions must be established before the workload can be distributed over the network. The policy vector for the determination of a site's user category restrictions is:

IPOLUA(ISITE,IUCAT,I)

where: ISITE = site number
IUCAT = user category
I = 1 or 2
1 - user category policy number
2 - parameter associated with the user category policy

3. Demand Allocation - After budget constraints have reduced demand as required and the user category restrictions have been imposed, the demand must be allocated among the available sites. For example, one of XYZ University's user category policies may be to limit all allocations to either itself or ABC. In this case, the demand allocation policies will be used to evaluate both sites and decide how much of the service type demand to allocate to each site. The current method involves a rating algorithm by which the sites are ranked and demands allocated in proportion to their rating. A variety of rating algorithms could be hypothesized. At present, the rating consists of a linear combination of price, turnaround, support, and past demand (momentum). The coefficients used in the rating equations for certain policies are site specific, e.g., a site can choose to look for good price and turnaround for one user category, and support for another. A site can assign a different coefficient to each rating component for each user category. The relative weights placed on the factors will determine where the demand for that user is allocated. The policy vector for the determination of a site's demand allocations is:

IPOLDA(ISITE,KTYPE,I)

where: ISITE = site number
KTYPE = service type
I = 1 or 2
1 - demand allocation policy set number
2 - parameter associated with the demand allocation policy set

The available service specific demand allocation policies currently include:

a. Look for the site offering the lowest prices. This policy is used by a site with a "cost conscious" profile, or in terms of overall demand practices, "price accountability."

b. Restrict network usage. Try to stay in-house as much as possible. This policy is used by a site with a "growth intensive" profile or with constraints on outside expenditures.

c. Look for the site offering the best combination of turnaround and price. This policy might be used by a site that is "user sensitive," but still has price accountability.

F. Market Policies

After all sites have allocated their demands for all user category levels and all service types, each site must check the feasibility of running all of the batch jobs, and supplying all the requested connect time. Factors to be considered include utilization of communications lines and other critical resources. The resource requirements for the total demand are calculated and compared to the available capacity. If the capacity for any critical resource is exceeded, the demand cannot be satisfied. If it is determined that a site cannot satisfy all demand requested for the week, the appropriate market (supply allocation) policy must be used to determine what demand will be satisfied and what unsatisfied. Some market policies currently implemented are:

1) For each resource that is over-utilized, the service types are cut in proportion to their usage of the over-utilized resource. Cut all sites equally independent of their usage (down to zero).

2) Cut back all work in proportion to the over-utilization independent of service type i.e., the job queue cannot

determine in advance which jobs to cut. The policy vector for the determination of a site's market practices is:

IPOLSA(ISITE,KTYPE,I)

where: ISITE = site number
KTYPE = service type
I = 1 or 2
1 - market policy set number
2 - parameter associated with the market policy set

G. Policy Conventions

A number of conventions relative to the representations of policies have been incorporated into the Network Simulation Model. The two most important, described below, concern the numbering of policies, and the time flags that permit the use of temporary or "trial" policies.

1. Overall Policy Conventions - The following conventions apply to policy numbers contained in the overall policy vector in the model:

Policy Number	Meaning
1. less than 0	Site chooses to implement a unique policy, (i.e., its own subroutine).
2. 0	No change in specific policy (i.e., same policy as last period).
3. greater than 0	Site chooses one of the standard policies available in the model. Substitute this policy number for whatever was used previously.

2. Second Level Policy Conventions - When dealing with second level policies the conventions are as follows:

<0 - site specific algorithm to be used
0 - do nothing, i.e., no action
>0 - the number of the standard algorithm to be used

3. Time Flags - Each policy has an associated time flag in the overall policy array, IOPOLC. Time flags are used so that a site may implement a policy other than its standard policy for any specified temporary period of time. After this time has elapsed, the site's standard policy will be restored. The current flags in use in array IOPOLC are:

	Time Flag	Meaning
1.	-1	New standard policy - save and set time flag to 999.
2.	0	Specified time has elapsed - restore standard policy and set time flag to 999.
3.	positive integer (n)	Policy will be used for n periods to follow, after which standard policy will be restored. (n is decreased by 1 each time period).
4.	999	Policy is to be used indefinitely. (Any integer greater than the number of time periods in the simulation run).

H. Representation of Budget and Cash Flow

Budgets are based on a yearly time interval, starting at a specified week (#1). The total bottom lines are not changed during the year (except in the "Exogenous Changes" module), but the dollar amounts allocated to the various categories can be reallocated using budget policies (practices).

Actual cash flows are configured in the same form as the budget representation. Monthly or weekly cash flows are projected to cover a yearly period in order to examine any discrepancies between

budgeted and actual amounts. The manner of projection (i.e., straight line this week; 1/52 annual; a function of total expenditures to date, weeks remaining, etc.) is a site-dependent function of policy. The vectors used for budget and cash flow are:

BUDGET(ISITE,ICAT) - the yearly budget

CHSFLO(ISITE,ICAT) - the actual cash flows (cumulative to date)

where: ISITE = site number
ICAT = income expense category

At the present time, BUDGET & CHSFLO have 25 income/expense categories each. These are as follows:

ICAT	CATEGORY
1	Total income of computer center
2	Total expenses of computer center
3	Internally generated income using school funds
4	Externally generated income, outside use
5	Other computer center income (grants, contributions, etc.) and/or deficit
6	Hardware/Software committed expense
7	Funds available for improvement
8	User support
9	Total user expenditures
10	Communications fixed expense
11	Communications variable expense
12	Supplies expense, cards, paper tapes
13	Operations staff
14	Programming staff
15	Administration expense
16	User category I expense
17	" " II expense
18	" " III expense
19	" " IV expense
20	" " V expense
21	" " VI expense
22	" " VII expense
23	" " VIII expense
24	" " IX expense
25	User category X expense